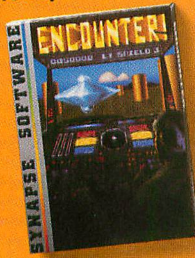

for Atari Computers
U.S.A. $3.00
CANADA $3.50
Antic®
The ATARI® Resource
JUNE 1984
VOLUME 3 NUMBER 2
EXPLORING THE ATARI XL
ACTIVISION INTERVIEW
Type-in game ESCAPE FROM EPSILON
NEW PRODUCTS
Solo Flight, Microfiler,
Telly Turtle,
and more . . .
ROBOTS, Part 3
06
0
74470 12728
ATARI—Official Home Computer of the 1984 Olympics
ATARI

SYNAPSE SATISFIES!



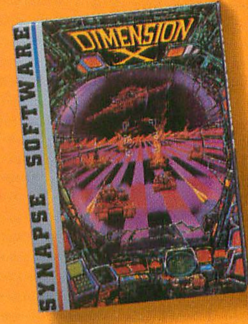
Encounter
Alone on a distant planet, you stumble into a futuristic 8-level tank battlefield that really plays for keeps! Watch out for the deadly drones and sneaky saucers.

Atari disk & cassette

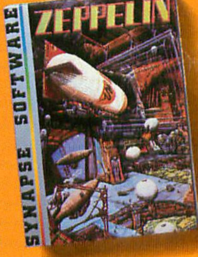


Dimension X
Test yourself with incredible 3-D action that never stops! An alien invasion fleet has the capital city surrounded, and it's up to you to stop the destruction! Good luck—you'll need it.

Atari disk & cassette



Zeppelin
First the prison break, but that's only the beginning! The underground world of Zarkafir is full of surprises, from the lethal energy fields to devastating earthquakes. Can you defeat the Timelords?



Atari disk & cassette

All Synapse entertainment titles are available at software dealers everywhere, or direct from Synapse for $34.95 plus $2 handling. Send check, money order or VISA/MasterCard number.

More Atari Titles:
Blue Max, Fort Apocalypse, Shamus and Shamus Case II, Necromancer, Pharaoh's Curse, Quasimodo.

Write for a FREE Synapse catalog. Synapse games are also available for the Commodore, Apple and IBM computers.

Name__________

Address__________

City__________

State__________ Zip__________

Computer__________ A



5221 Central Avenue, Richmond, CA 94804 • 415/527-7751



THE GREATEST ATARI® GAME OF ALL TIME.

Object: Capture more programs than from any other source.
Score: The best prices for programs win.

Actually, every member of CompuClub is a winner, because no one pays lower prices for Atari® programs than our members.

And no other source offers as many programs, with a descriptive catalog covering every piece of software we offer!

CompuClub has hundreds of Atari® programs: games, education and business. It's an astonishing selection, but just as incredible are our prices and our catalog.

Prices always at least 25% below retail

Fact is, usually our prices on selected programs are even lower than that. And we're always running sales with savings of at least 50% from the list price on some of your favorite programs.

And there are several ways to pay: MasterCard, Visa, or the ever popular check or money order.

Annotated catalog, updated every 45 days

We don't like to knock the competition, even by implication, but there are definite advantages to a CompuClub membership. One of the best of them is our catalog, which not only offers an unbelievable number of programs, but includes a description of each and every one of them.

In fact, the catalog is so big and filled with so much information that we print a separate price list. And we keep adding so many programs that we have to update the catalog nine times a year during the course of your membership.

Think of it! No more wondering what's behind the fancy label, no more shooting in the dark or depending on the word of a clerk who may not be familiar with the program you're interested in.

Exclusivity for Atari® owners

We're not trying to be all things to all people. We are definitely the best thing since the floppy disk for Atari® owners, and only Atari® owners. CompuClub is very exclusive.

Our catalog is thick with hundreds of Atari® programs, and only Atari® programs. No more fumbling your way through thickets of strange symbols and codes for the different computers everybody else's catalogs try to cover.

The rules

The rules are simple. To play the CompuClub "game," you've got to be a member. Anyone can join, as long as they fork over a mere five bucks, and agree to buy three programs during the year of membership. Order and buy your programs at any time during the year, but we're sure with our selection and prices that you'll want to get going right away.

Five dollars buys you a one-year membership, exclusive Atari® program offerings, a fully annotated catalog with 9 updates during the year, a current price list (and any necessary updates), a subscription to our newsletter, and discounted prices at all times, including announcements of our periodic sales, which feature savings of 50% off and more.

You can join by filling out the coupon below, or get an instant membership by calling our toll-free number and giving us your MasterCard or Visa identifying number. We'll forward your registration and sales order materials, your catalog and price list, so you can begin ordering right away.

That's all there is to it.

If you think you play a good game, you've got to join CompuClub. It's easy to play, and it saves you money.

We think it's the best Atari® game going, because with us, everybody wins.

CompuClub
Where Atari owners belong.



YES, **CompuClub,** PO Box 652, Natick MA 01760 **ATTN: Dept. AT**
I want to play the greatest Atari® game of all. Please enroll me in CompuClub for one year, and rush my registration material, catalog and price list to me. I agree to buy any three programs during the course of my year of membership. I have enclosed my payment of $5, or authorization to bill my charge card account.

Please make your check payable to CompuClub
Payment enclosed ☐ check ☐ money order.
Bill my ☐ MasterCard ☐ Visa ID #__________
Expires__________
Signature__________
Name__________
Address__________
City State Zip
Computer Model__________ Disk ☐ Tape ☐

If you are not satisfied in any way with the conditions of your membership, you may cancel before ordering any programs from CompuClub. Your membership fee of $5 will be cheerfully refunded.

To join by phone, call toll-free.
800-631-3111

Please have your MasterCard or Visa card ready when you call. In Massachusetts, call 1-617-879-5232. We will forward your membership materials after confirming authorization for your charge.

Boot up a brainstorm.

You're going to need all your cunning and intelligence to stop the FuzzBomb from turning the U.S.A. into a nation of fuzzbodies.

You're Agent U.S.A.,™ top investigator for the country's most elite intelligence organization. And now you've been assigned your toughest and most dangerous mission ever: stopping the evil FuzzBomb from turning every man, woman and child into helpless fuzzbodies.

Quick reflexes and sharp eyes are not nearly enough here. As you race around the country on super-fast rocket trains, you'll need all your smarts to outwit the devious plague spreader. But watch out! One wrong move and you're just another fuzz-body.

And once you've tested your mettle against the FuzzBomb, you'll be ready for more mind-stretching Wizware.™

Bannercatch.™

In a remote field bigger than any field you can imagine, you and a friend team up to take on chief robot Max and his crack robot raiders. Infiltrate their territory and grab their flag before they grab yours. It may sound simple, but your strategy better be good. Max has lots of surprises in store for you.

Spelldiver.™

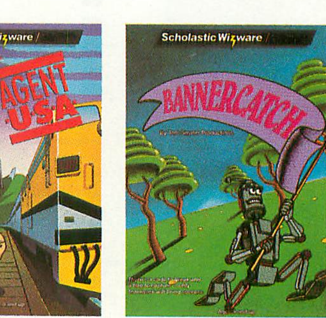
You're a scuba diver on a secret assignment. You must find and decipher giant words hidden deep beneath the sea. But to succeed you have to out-maneuver ferocious sharks and pesky flipper-nippers that are out to stop you.

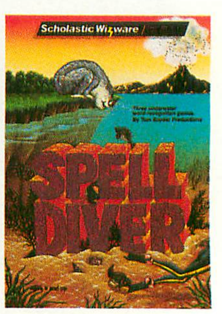
So get ready for challenge and excitement. Boot up Wizware and let the brainstorm begin.

Look for Scholastic Wizware at your local computer store. Or contact Scholastic Inc., 730 Broadway, New York, NY 10003, 212-505-3000.

For a free red-and-white "Boot Up a Brainstorm" T-shirt, send your name, address, shirt size and computer brand to "Boot Up" c/o Scholastic at the above address. Enclose $2.00 for shipping and handling.

Scholastic™ Wizware





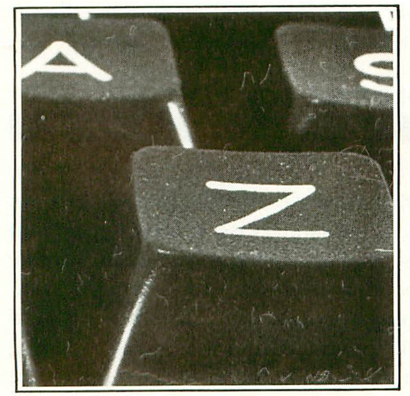
Agent U.S.A., Bannercatch, Spelldiver designed and developed by Tom Snyder Productions, Inc. Available for Atari and Commodore 64 computers. Apple and IBM versions available soon.



The ATARI Resource JUNE 1984 VOLUME 3, NUMBER 2



page 16



page 38



page 71

FEATURES

DEPARTMENTS



Publisher
James Capparell

Editorial Department
Robert DeWitt, Managing Editor
Christopher Rauber, Assistant Editor
David Duberman, Technical Editor
Caitlin Morgan, Editorial Assistant
Arla Ertz, Copy Assistant
Jack Powell, Technical Assistant

Contributing Editors
Carl Evans, Ken Harms,
David & Sandy Small, Jerry White

Art Department
Marni Tapscott, Art Director
Kyle Bogertman, Production Supervisor
Linda Tapscott, Ad Production Coordinator
Patricia Fostar, Production Assistant

Contributing Artists
Beatrice Benjamin, Linda Tapscott,
Pete McDonnell

Cover Illustration
Bud Thon

Circulation Department
Les Torok, Manager
Augustus Jones, Shipping & Sales
Peter Walsh, Shipping
Julianna Hoffman, Subscriptions
Monica Burrell, Subscriptions

Accounting Department
V.J. Briggs, Manager
Saundra Jones, Accounting Clerk
Brenda Oliver, Accounts Receivable

Marketing
Gary Yost, Manager, Marketing Support
Steve Randall, Advertising Sales Director

General Offices (415) 957-0886
Advertising Sales (415) 661-3400
Credit Card Subscriptions
outside California (800) 227-1617 ext. 133
inside California (800) 772-3545 ext. 133
Subscription Problems (415) 957-0886

June 1984
Volume 3, Number 2
ANTIC—The ATARI Resource is published twelve times per year by Antic Publishing. Editorial offices are located at 524 Second Street, San Francisco, CA 94107. ISSN 0745-2527. Second Class Postage paid at San Francisco, California and additional mailing offices. POSTMASTER: Send address change to **Antic**, 524 Second Street, San Francisco, CA 94107.

Editorial submissions should include program listing on disk or cassette, and text file on media and paper if text was prepared with a word processor. Media will be returned if self-addressed stamped mailer is supplied. **Antic** assumes no responsibility for unsolicited editorial material.

Antic is an independent periodical not affiliated in any way with Atari, Inc. ATARI is a trademark of Atari, Inc. All references to Atari products are trademarked and should be so noted.

i/o board

DUE TO OVERWHELMING DEMAND . . .

*In our last issue (**Antic**, I/O Board, p. 10, April 1984) we indicated that we were planning to publish some disk drive reviews in the June issue. In order to cover disk drives appropriately, we've decided to postpone publication of the survey until August, and to make disk drives the theme of that issue. Many of you are anxious to see the survey, and we thank you for your patience—we're doing our best to make it worth the wait.*
—ANTIC ED

A GAME FOR FOUR

I'm writing to request that you publish a multiple-player game. My wife and I enjoy games that we can play with our friends, and we often play against each other. Multiple-player games allow the whole family to play at the same time.

Mike Lewis
Salt Lake City, UT

We thank you for your suggestion, and call your attention to "Tank Battle," the four-player game in this issue. Have fun with it!
—ANTIC ED

ATR-8000

I'd like to know if it's possible to use a hard disk with the ATR-8000 CP/M expansion system for the Atari. I'm also interested in opening a CP/M-based bulletin board; do you know of any good CP/M bulletin-board programs?

Mike Widner
Toms River, NJ

The following response was provided by David Small:

Fred Helms, Marketing Director for SWP (the makers of the ATR-8000), tells me that there is a BBS that has been customized for the ATR-8000. It is called FORUM, and is available from Matt Singer, (301) 871-6131, 2405 E. Gate, Silver Springs, MD 20906. Or call Fred at (817) 469-1181.

I feel that you will enjoy a CP/M-based bulletin board program because of the rapid disk-response time and large storage sizes possible with the ATR.

While SWP does not offer a hard disk for the ATR, there has been considerable interest in bringing one up. There are a few independently designed hard disks currently running on ATR's; however, whether or not they will ever be marketed is another question.

TRACE TROUBLE

I've discovered an apparent bug in the Atari Assembler Editor cartridge that I haven't seen documented anywhere. If, when performing a Trace (note: name of function) in the debugger, a CPY instruction is encountered, this causes the trace to be aborted and control to be returned to the debugger. Sometimes the system crashes when this happens. Do you know anything about this?

Paul Mattia
Sterling, WA

*We confirmed Paul's finding here at **Antic**. If you have any suggestions or comments about this, please send them to I/O Board.*
—ANTIC ED

ROLL CALL FOR ROBOTS

In your December and January issues, you carried articles on building a robot by Evan Rosen. The January issue referred to a Forth pulse routine, similar to the BASIC routine listed, that would be published in the forthcoming February issue. Yet when I looked for it, I couldn't find it! Are you planning to publish any more of Rosen's wonderful articles? What's the scoop?

Jerry D. Lumpkins
Christye L. Robley
New Orleans, LA

*A number of readers have written with the same questions. As noted last month (**Antic**, HELP!, p. 12, April 1984), a combination of factors prevented us from continuing Evan Rosen's robot series in the February issue. However, see "Talk to Your Robot" in this issue for more on robotics from Rosen.*—ANTIC ED

ANTIC'S NEW SCHEDULE

*In case you hadn't noticed, **Antic** did not publish a May issue this year!*

We didn't skip an issue, however! We simply moved up our cover date to facilitate distribution of the magazine. As a result, this issue is reaching you several

i/o board

weeks earlier than usual.

This change will not affect your subscription or the number of issues published during 1984: There will be 12 issues in this volume, and all subscriptions will be adjusted automatically to reflect this change. You can expect to receive each future issue at least two weeks before its cover date.
—ANTIC ED

CREDIT DUE

We were very happy to see the positive review of **MovieMaker** in your April issue. We use it extensively here at the Computer Arts Forum in our computer-graphics classes because it is an excellent product. However, as a point of clarification, **Antic** contacted us, not Interactive Picture Systems, regarding the production of the animated Xmas card (which was directed, produced, and coordinated by our facilities). IPS did contribute the "Merry Christmas" sequence to the Xmas card, and provided inestimable support during the project's development. It is our purpose to actively encourage the use of microcomputers in graphics production, and we appreciate the fact that **Antic** has taken such an innovative and supportive role in this process.

Susan Bickford, Director
Computer Arts Forum,
Pratt Institute
New York City

ATARI'S WAY WITH WORDS

Thank you for your fine article "Atari and Epson" by Douglas MacKay (**Antic**, January 1984). Though I own an Epson MX-80, I usually use AtariWriter with my Dynax-15 for word processing. Luckily, with the help of Mr. MacKay's article and the Dynax instruction manual, a number of heretofore unavailable print functions are now at my command.

Incidentally, I constantly read that the Atari is a poor word processing machine. Well, rubbish to that! I use my machine five hours a day for correspondence and managing forty-five employees. I couldn't find a better, more reliable machine.

Lynn W. Sabin
Camp Pendleton, CA

PRODUCING VIDEO TITLES

In response to E. Benchimol's question in the March "HELP!" column, I've used my Atari 800 as a video titler on several occasions.

To do this, write a BASIC program in Graphics 1, 2, or 3 that provides the desired titles. See the listing below for an example. When you're ready to record your titles, simply hook the switch box (CA010112) to the antenna input on the video tape recorder, instead of on the television. Then set the video tape recorder to record, and run the program.

```
10 REM TITLE SAMPLER
20 GRAPHICS 2+16:FOR DELAY=1 TO
   1000:NEXT DELAY
30 POSITION 6,4:PRINT #6;
   "TITLES BY"
40 POSITION 3,6:PRINT #6;
   "JOHN SKOVMAND"
50 FOR DELAY=1 TO 1000: NEXT
   DELAY
60 GRAPHICS 2+16
70 GOTO 70
```

I've been reading **Antic** ever since I bought my Atari 800, and I look forward to every issue—keep up the good work.

John Skovmand
Ventura, CA

FREAKOUT

Here is a short "mood" program for your readers. We call it "FREAKOUT."

```
10 DIM A(18):GRAPHICS 18:
POKE 756,4:FOR I=1 TO 18
:READ X:A(I)=X:NEXT I
20 L=INT(RND(0)*18)+1:FOR
 REG=0 TO 3:FOR VOL=1 TO
 7:SOUND REG,A(L),12,VOL:
NEXT VOL:POKE 1024+REG*2
,PEEK(53770)
30 FOR I=1 TO 65:NEXT I:
POKE 1025+REG*2,PEEK(537
70):NEXT REG:POKE 708,PE
EK(53770):POKE 712,PEEK(5
3770):GOTO 20
40 DATA 243,217,193,182,
162,144,128,121,108,96,9
1,81,72,63,60,52,48,45
```

Clark and Mahlen Morris
Orinda, CA

THE ENGLISH CONNECTION

As England's major independent producer of software for the Atari computers, we find **Antic** to be an invaluable source of technical information which aids us in our development of games and utility software for the best home computers! Having now produced software for nearly two years, we are keen to set up a dialogue with American programmers, with a view to the exchange of technical information and programming aids. To this end, we would welcome correspondence to the address below. Keep up the good work!

Philip Morris
English Software
Box 43
Manchester M60
England, UK

A MESSAGE-DISPLAY PROGRAM

In answer to the question by E. Benchimol in your March "HELP!" column, my MESSAGE DISPLAY PROGRAM can create video titles as well as display messages over cable TV systems. It is currently used for both purposes by many schools, colleges and small cable operators. In the Philadelphia area, for example, five school systems use it over their educational access channels. In many cases, Atari computers have been running it 24 hours a day for almost two years. A good description of the program appeared in the March 1983 issue of *Educational & Industrial Television Magazine*, and we'd be happy to send a reprint to anyone interested in a copy.

Dennis J. Harkins
Harkins Associates
144 S. Limekiln Pike #4
Chalfont, PA 18914

MORE ON THE XL

I enjoyed your February 1984 issue, and found your product reviews to be very helpful and fairly complete. But I was dismayed to find very little information regarding the Atari XL series computers. You referred to Antic Mode E (Graphics 7.5). On the XL series, this mode is built into BASIC as

continued on next page

i/o board

Graphics 15. I am sure that many new owners (and readers of your magazine) do not know that these are one and the same.

Overall, you do a fine job of covering the Atari computers. In fact, I typed this letter on AtariWriter, which I bought after reading your review. Keep up the good work!

Kevin Crossman
Palo Alto, CA

You'll find a closer look at Atari's new XL line in this issue.—ANTIC ED

GEMINI 10X

Thank you for spotlighting our product, the Gemini 10X printer, in your January 1984 issue. However, I must draw your attention to some errors. First, the table of printer comparisons on page 55 states that the Gemini has no buffer. In fact, the Gemini has an 816-byte buffer, which is expandable to from 4K to 8K. Also, the description of the Gemini on page 56 states that it is "... significantly slower than its rated 128 characters per second." The correct figure is 120 characters per second.

Eric J. van Hall
Product Manager
Star Micronics, Inc.

Thank you for the corrections, Mr. Hall. The Gemini 10X that you so graciously donated to ANTIC is one of our most often-used printers. It has functioned flawlessly for us. —ANTIC ED

DOUBLE TROUBLE

I use both the Make a Face printer interface (ANTIC, October 1983, page 53) and GTIA Sketchpad (ANTIC, December 1983, page 137). However, I'm having trouble using them together. How can I modify either program to get a printer dump of the GTIA Sketchpad screen?

Dave Carpenter
Pitcairn, PA

GITA Sketchpad wipes out the Make a Face machine code in Page Six, so the code must be relocated. The best solution, if you have an assembler, is to type in the source code on page 56 of the October issue, and reassemble at a different address. The only changes in the resulting object code will be the addresses for the JSR instructions in lines 650 and 1370. We suggest that you find the start of screen memory for GTIA Sketchpad (33104 with 48K), and set the new origin about two hundred bytes back from this. Once you've reassembled the Make a Face routine, you can write a small BASIC program to POKE the routine into its new location, and POKE the new address into locations 795 and 796. For instance, if you set the new origin at 32900, you'd POKE 795,132 and POKE 796,128. —ANTIC ED

help!

EEEK! A BUG!

Readers of **Antic** often use TYPO (a debugging program last published in the February 1984 issue) when typing in programs from the magazine. TYPO requires that you LIST a version of the program to disk or cassette, type NEW, and then ENTER the program to clean up the variable name table.

Early versions of the Atari 600XL and 800XL contain Revision B of Atari BASIC ROM, which has a small bug. Newer machines should contain Revision C, which is bug-free.

This bug can cause problems when you use TYPO, because certain LISTed files cause the computer to lock up when you attempt to ENTER them. Therefore, you should always SAVE a file *before* you LIST it to the storage medium.

If you encounter lock-up when you try to ENTER a file, LOAD the SAVEd version and change one byte near the beginning of the file. For instance, if there's a REM line near the beginning of the file, add or delete a character. If there's not a REM line near the beginning, add a REM to the end of the first line. If you then LIST this file to a disk or cassette, you should be able to ENTER it with no problem.

If you have only the LISTed version of the program, you can ENTER it only by plugging a BASIC cartridge into your computer. The Translator program will not work in this case.

See "Exploring the XL" in this issue for further information on the three versions of Atari BASIC.

GTIA FIX

When using "GTIA Sketchpad" (ANTIC, December 1983), I found that you can only fill an enclosed area with the border color. For example, if you draw a box with Color 4, you can only fill it with the same color. If you try to use a different color for fill, the fill goes beyond the borders. Here's my fix for this problem. Change:

IF COLR=C

in lines 250, 270, 300, and 340, to:

IF (COLR>0)=(C>0)

This change allows you to fill an enclosed area with any color. You can also fill with background color to erase objects.

Scott Sheck
Gaithersburg, MD

CASSETTE WARNING

In the October 1983 issue of **Antic**, we published a letter and a short program listing entitled "Reset to Rerun" in I/O Board. When combined with another program, this routine causes the program to RUN again automatically if [RESET] is pressed during the program run. The routine does not work correctly on a cassette-only system unless you POKE 9,1. Add the POKE statement to the end of line 0.

OUR MISTAKE

While I do not regret upgrading my 400 to 48K (something I had long intended, but always postponed), it was my desire to utilize Epyx's "Fun With Art" that prompted the move. When I first spotted "Fun With Art," it required 32K. So I had the upgrade installed, and then purchased the cartridge. I have enjoyed using it ever since.

My reason for writing is your February 1984 review of "Fun With Art," which listed it as requiring only 16K! Do two versions exist?

John Donahue
N. Miami, FL

No. There is no 16K version of "Fun With Art." Our review included an incorrect figure; it should have listed the RAM requirement as 32K.—ANTIC ED

HELP! Yourself

Atari maintains toll-free telephone assistance numbers in the continental U.S.

Hours (Pacific Standard Time)
7 a.m. to 11 a.m. —
noon to 4 p.m.

Within California call
(800) 672-1404

Elsewhere dial
(800) 538-8543

SHIPPING IS EXTRA

In "ANTIC Pix Gifts" (ANTIC, page 118, November 1983), we stated that the $89.95 price of Hytec Systems' CS 1632 cabinet included the shipping fee. This is incorrect. Shipping charges are extra, and depend on the destination involved.

OH, THOSE O'S

I'd like to point out a problem in your magazine listing of PM Mover. The zeroes are shown as the letter "O" (inverse O).

William P. Schneider
Kenner, LA

*We've checked the program listing that accompanied "Automate Your Player/Missiles" (**Antic**, p. 66, December 1983) and indeed:*

Line 1250 contains two inverse O's that should be inverse zeroes, and line 1265 contains one inverse O that should be an inverse zero. All other inverse O's are correct.

Since this is part of a machine-language string, the program will not operate properly until fixed.—ANTIC ED

ESCAPE MAZE

I found a flaw in "Escape Maze" (**Antic**, p. 86, March 1984). If you run into the wall on Maze 1 four times, the game starts over (line 1080). After that, no matter how many times you run into the wall, all that happens is that you're sent back to start; the variable HIT doesn't change back to 0. But if you add the following line:

75 HIT=0

it will work every time.

I'm glad that you include the Kbyte requirement for each game, and I compliment you on a great magazine.

John Marchant
Racine, WI

STRIKE THREE!

In our recent review of Gamestar Software's Star League Baseball (**Antic**, p. 100, March 1984), we erroneously reported that the program required 48K. It actually requires only 32K. We apologize for this error.

EVOLUTION OF THE XL COMPUTERS

Memory and expandability tell the tale

by ROBERT DEWITT
Managing Editor

The XL line of Atari computers has hit the market. Hundreds of thousands of the gleaming machines are now in the hands of eager Atarians. Most of the new owners are new to computing, but some have bought XLs to replace or supplement an earlier unit. Everyone seems to be curious about the powers and peculiarities of these new computers.

Antic has at least one of each of the XLs in service: the 600XL, the 800XL and the 1200XL. To broaden our perspective, we also visited recently with members of the Atari engineering staff and inquired about the evolution of these computers from the 400 and 800 models with which most of us started.

The key notion for development at Atari has been "upward" compatibility. This means that old software should work with new models, but new software for new models may not work with old models. So far, all Atari computers operate in essentially the same way, based on the 6502 eight-bit microprocessor supported by several other chips including ANTIC, GTIA (CTIA in early models), POKEY, and PIA.

Software that is properly written for the Atari 400 and 800 computers *will* run on the XL machines. Unfortunately, some software products, including some Atari products, were not properly written and will not work with the XLs. Among the incompatible titles of which we are aware are **Atari Word Processor** (not **AtariWriter**), **Text Wizard**, and LJK's **Letter Perfect**.

One major hardware change introduced with the XL machines does create some problems. Only two joystick ports are now available, as opposed to four with the previous models. Any program that uses all four ports at once will not work properly on the XL machines, and any peripheral that uses Ports 3 and 4 cannot be connected.

It is, in fact, the deletion of these ports that made possible many of the added features of the XL machines. In all Atari computers, the PIA chip handles the joysticks. In the 400 and 800 this was accomplished through addresses 54016 (called PORTA) and 54017 (called PORTB). PORTB formerly handled joystick Ports 3 and 4. In the XLs, PORTB is devoted to other functions: switching the Operating System in and out, switching BASIC in and out, and invoking self-tests. Note that the 1200XL does not have built-in BASIC, so programmable LEDs on the console are controlled here instead.

In the XL computers, programs that call for the alternating use of four joysticks are supported by software simulation of the third and fourth sticks. Programs with four independently movable players will show Players 1 and 3 being moved simultaneously by Stick 1 and Players 2 and 4 being moved by Stick 2.

CHEAP MEMORY

When the Atari 400 and 800 were developed, memory chips were more expensive, less capable, and required more power. Early models had as little as 2K RAM. Now 64K memory chips are widely available. This is one of the technological developments that gave rise to the XL machines. (Strangely enough, the Atari 400 was always considered complete [by Atari] at 16K RAM, even though other companies discovered ways to expand it.)

The Atari 800 was designed with accessible slots explicitly so that its memory could be expanded to 48K RAM. Atari did not forsee any of the other uses eventually made of these slots, nor did it even anticipate that full memory was possible with fewer than three RAM boards. The machine was rated at 48K not because it was incapable of using more RAM, but because 48K seemed to be "enough" at the time, and it was deemed reasonable by Atari to set aside 16K for housekeeping and future expansion.

One portion of the housekeeping area is of special importance. This is the 2K range of addresses (from $D000 through $D7FF) known as the I/O block. These addresses serve the several support chips mentioned above, and have other functions that preserve compatibility among all of the Atari computers. This area can never be used as RAM, not even when the OS is switched out.

Why would you want to switch out your Operating System? Credit the horsepower race for this concept. The Commodore 64, also a 6502-based

Table 1

XL MEMORY MAP

The following memory map assumes that DOS 2.0S is booted with a Graphics Mode 0 screen in BASIC.

HEXADECIMAL*	DECIMAL	USE
0000–007F	0–127	OS page zero RAM
0080–00FF	128–255	user page zero RAM (BASIC)
0100–01FF	256–511	6502 stack
0200–05FF	512–1535	OS RAM
0600–06FF	1536–1791	FREE RAM
0700–1CFB	1792–7419	DOS
1CFC–9C1E	7420–39966	user RAM (BASIC)
9C1F–9FFF	39967–40959	display list and screen RAM
A000–BFFF	40960–49151	BASIC cartridge
C000–CBFF	49152–52223	OS ROM
CC00–CFFF	52224–53247	OS ROM (intn'l character set)
D000–D0FF	53248–53503	GTIA registers
D100–D1FF	53504–53759	reserved for future use
D200–D2FF	53760–54015	POKEY registers
D300–D3FF	54016–54271	PIA registers
D400–D4FF	54272–54527	ANTIC registers
D500–D7FF	54528–55295	reserved for future use
D800–DFFF	55296–57343	OS ROM (floating point package)
E000–E3FF	57344–58367	OS ROM (domestic character set)
E400–FFFF	58368–65535	OS ROM

*The Hexadecimal number system uses 16 as its base.

Table 2

OS CHANGES FROM REV. B TO THE XL

This is a list of memory differences between the Rev. B Operating System of the 400/800s and the Operating System developed for the XL computers.

HEX ADDRESS	REV. B USE	XL USE	DESCRIPTION
0000	reserved	LNFLAG	— for inhouse debugger
0001	reserved	NGFLAG	— for power-up selftest
001C	PTIMOT moved (0314)	ABUFPT	— reserved
001D	PBPNT moved (02DE)	"	
001E	PBUFSZ moved (02DF)	"	
001F	PTEMP (deleted)	"	
0036	CRETRY moved (029C)	LTEMP	— loader temp.
0037	DRETRY moved (02BD)	"	
004A	CKEY moved (03E9)	ZCHAIN	— handler-loader temp.
004B	CASSBT moved (03E9)	"	
0060	NEWROW moved (02F5)	FKDEF	— func key def ptr.
0061	NEWCOL moved (02F6)	"	
0062	" (02F7)	PALNTS	— PAL/NTSC flag.
0079	ROWINC moved (02F8)	KEYDEF	— key def pointer
007A	CLINC moved (02F9)	"	
0233	reserved	LCOUNT	— loader temp
0238–0239	"	RELADR	— loader
0245	"	RECLEN	— "
0247	LINBUF (deleted)	reserved	

continued on page 14

computer and a direct Atari competitor, claimed to be a 64K machine. Of course, the Atari 800 was one, too, but it provided a maximum of only 48K RAM. Atari's solution, first implemented with the 1200XL, was to install a 64K "underlay" of RAM chips. This would be available (except for the ever-present and protected I/O block) for whatever use the programmer desired, including use of a different Operating System. This handy feature allows regression to the old "Rev. B" OS that is needed to run incompatible older software. Atari sells this OS as a "Translator Disk" for $9.95, and may soon offer it on cassette.

THE NEED FOR THE 1200XL

The 1200XL project offered several other opportunities. Exterior redesign of the Atari computer was urgently needed. The beige beasts of the past, attractive in their way, just didn't have the high-tech look. The XL line would shoot for elegance — it would be coffee and cream trimmed in silver, with a sleek, low profile made possible by its single-board design.

The 400 and 800 were multi-board computers. But the 1200XL needed only a single board. Technology had improved enough in just a few years to reduce the number of required integrated-circuit chips by half. With almost everything mounted on one sturdy printed-circuit board, the need for connectors and problems with connectors were reduced dramatically. Cartridges the size of Atari's standard BASIC could now carry 16K. When the old 800 was designed, a future need for 16K (for cartridge-based external programs) was provided for by the right-hand cartridge slot (which accesses the 8K of memory directly below BASIC's 8K). The new cartridge made this little-used slot completely obsolete, however, so the 1200XL included just one cartridge slot, which could access either the whole 16K at the top of RAM or only 8K, if that was all that was needed by the

continued on page 14


ATARI 850
The TRAK AT-D4
is compatible with
all Atari
Computer Systems
Atari is a registered trademark of Atari, Inc.

EVOLUTION OF THE XL COMPUTERS
continued from page 11

cartridge.

The 1200XL was considered to be a complete computer. No attempt was made to expose the parallel bus, at least in part because the engineers were wary of the "crazy" things that might be plugged in there. No extra slots were opened because the unit came with all the RAM it could ever use. Well, that concept of architecture simply foreclosed the future. The hobbyists and even the mass consumers seemed to balk at the machine's limitations, especially at its price tag.

THE NEW MACHINES

The next machines, the 600XL and the 800XL, would address those grievances. Styled similarly to the 1200XL, they would expose the parallel bus for whatever uses an inventive public could devise, and would carry built-in BASIC at a price that would attract the multitudes.

The 600XL and 800XL are of similar design, but the 600XL has only 16K RAM installed, and drives only a television, not a monitor. A RAM-expansion module is planned for the 600XL. This will continue to pass the bus to other add-ons. Among these will be an expansion box from Atari that will accept, Apple IIe style, third-party boards of sundry kinds, thus opening the way for the unlimited development of the XL machines.

The power supply for the 600XL and 800XL is external. As a result, they provide, for the first time in the history of Atari computers, a single 5-volt DC current. Another distinction of the XL computers is that they feature a true hardware reset, as opposed to a software interrupt. Other XL doodads include programmable control of keystroke rate and clicks, keyboard disable, fine scrolling in text mode, programmable function keys (1200XL only), and an international character set in ROM.

The Operating System for Atari computers has gone through an evolution as well. PEEK(65527) returns a value that identifies the resident OS. The 400/800 Rev. A=221; 400/800 Rev. B=255; 1200XL Rev. A=10; 1200XL Rev. B=11; 600XL=1 and 800XL=2. The major divergence is between the OS for the 1200XL and Rev. B for the 400/800s. A gross memory map for the XL computers is shown in Table 1. A list of significant address changes is shown in Table 2. This information was derived from the recent but undated "Guidelines," a document prepared by Atari, Inc., for those seeking detailed information about recent changes.

Table 2 continued

HEX ADDRESS	REV. B USE	XL USE	DESCRIPTION
0248-026A	"	reserved	
026B	"	CHSALT	— character set ptr
026C	"	VSFLAG	— fine scroll temp
026D	"	KEYDIS	— keyboard disable
026E	"	FINE	— fine scroll flag
0288	CSTAT (deleted)	HIBYTE	— loader
028E	reserved	NEWADR	— loader
029C	TMPX1 (deleted)	CRETRY	— from 0036
02BD	HOLD5 (deleted)	DRETRY	— from 0037
02C9-02CA	reserved	RUNADR	— loader
02CB-02CC	"	HIUSED	— loader
02CD-02CE	"	ZHIUSE	— loader
02CF-02D0	"	GBYTEA	— loader
02D1-02D2	"	LOADAD	— loader
02D3-02D4	"	ZLOADA	— loader
02D5-02D6	"	DSCTLN	— disk sector size
02D7-02D8	"	ACMISR	— reserved
02D9	"	KRPDEL	— auto key delay
02DA	"	KEYREP	— auto key rate
02DB	"	NOCLIK	— key click disable
02DC	"	HELPFG	— HELP key flag
02DD	"	DMASAV	— DMA state save
02DE	"	PBPNT	— from 001D
02DF	"	PBUFSZ	— from 001E
02E9	"	HNDLOD	— handler loader flag
02F5	"	NEWROW	— from 0060
02F6-02F7	"	NEWCOL	— from 0061
02F8	"	ROWINC	— from 0079
02F9	"	COLINC	— from 007A
030E	ADDCOR (deleted)	JMPERS	— option jumpers
0314	TEMP2 moved (0313)	PTIMOT	— from 001C
033D	reserved	PUPBT1	— power-up/reset
033E	"	PUPBT2	— "
033F	"	PUPBT3	— "
03E8	"	SUPERF	— screen editor
03E9	"	CKEY	— from 004A
03EA	"	CASSBT	— from 004B
03EB	"	CARTCK	— cart checksum
03ED-03F8	"	ACMVAR	— reserved
03F9	"	MINTLK	— "
03FA	"	GINTLK	— cart interlock
033FB-03FC	"	CHLINK	— handler chain

CP/M and MS-DOS Compatibility For All Atari Computers







"A fine CP/M machine . . . the ATR8000 closes the gap that has separated Atari owners from the rest of the software market."

—*Byte Magazine*

CP/M compatability for your Atari — only from SWP.

Now you can run CP/M programs on your Atari 400, 800, 1200 and XL series computers.

The ATR8000 from SWP Microcomputer Products is a 4MHz, 16k RAM complete Atari interface. When you add the optional 64k RAM, the ATR8000 will run CP/M!

The ATR8000 features:

- 64k RAM
- RS-232 port with software to run a serial printer or modem.
- The ability to run up to four 5 1/4" or 8 " drives of any mixture, type and density (MY-DOS is optional and is used for double density Atari operations).
- Double density CP/M 2.2.
- 80 column wide display with 40 column moveable window. Or an optional 80 column software (AUTOTERM-80) program that can be used with a black & white TV or monitor.

When you order your ATR8000, you'll not only receive the hardware and software you need, but also an easy to read owner's manual and CP/M supplement that will take you into the exciting world of CP/M.

SWP

MICROCOMPUTER PRODUCTS, INC.

2500 E. Randol Mill Rd., Suite 125, Arlington, TX 76011

Add MS-DOS compatibility too!

An Atari that runs MS-DOS, IBM-PC software?

By adding CO-POWER-88, our own 8088 co-procesor to your ATR8000, you can also run most of the popular MS-DOS, IBM-PC software.

The CO-POWER-88 is a 16 bit, 5.33 MHz, 8088 co-processor with 128k or 256k RAM. It comes complete with MS-DOS and RAM disk software. CP/M-86 is available as an option.

More than just a co-processor, its RAM can also be used as a high speed simulated disk drive for CP/M.

Installation is simple. Your CO-POWER-88 comes with the Z-80 adaptor board and main processor board. You'll also receive a complete owner's manual and easy-to-follow instructions.

CP/M and MS-DOS, IBM-PC compatibility for your Atari can now be yours.

Contact your local SWP dealer or order direct by mail. Better yet, for fastest delivery, use your credit card and shop by phone.

Credit Card Orders Save Time!
Call 817-469-1181 or 817-861-0421 And Order Now
Or Mail Coupon to: SWP, 2500 E. Randol Mill Rd., Sutie 125, Arlington, TX 76011

Please ship the following:

☐ 16k ATR8000 Atari Interface @ $350.00 + $10.00 shipping.
☐ 64k ATR8000 w/CP/M @ $499.95 + $10.00 shipping.
☐ 128k CO-POWER-88 with MS-DOS @ $400.00 + $10.00 shipping. *
☐ 256k CO-POWER-88 with MS-DOS @ $500.00 + $10.00 shipping. *
☐ CP-86 (16-Bit Disk Operating System) @ $60.00 + $5.00 shipping.
☐ MY-DOS (Atari-like Disk Operating System) @ $29.95 + $2.00 shipping
☐ AUTOTERM-80 (80 column display for CP/M or MS-DOS) @ $29.95 + $2.00 shipping
* Must be used with 64k ATR8000.

Specify Atari model computer you own: ____________
Configurations: (check one) 5 1/4" drive ____ or 8" drive ____
☐ I want to learn more about the ATR8000 and CO-POWER-88. Enclosed is $1.00, (no checks please) for more literature, including a comprehensive listing of MS-DOS and CP/M-86 software that runs on the CO-POWER-88.

Method of payment:
☐ Personal Check enclosed (must clear bank before shipping)
☐ Certified Check or Money Order enclosed.
☐ Charge the following credit card:
☐ Master Card ☐ Visa
Credit Card # ____________
Master Card Bank # ____________ Exp. Date ____________
Subtotal: ________ + Shipping ________ + Sales Tax (Texas Residents add 5%) ________ =
Total Enclosed: ________
Name ____________
Address ____________
City ________ State ________ Zip ________
Phone (h) (____)____-______ (w) (____)____-______

Signature ____________



ALPHABET MUSIC

A musical way to learn the ABC's

by RICHARD SELTZER

Alphabet Music is a game that introduces preschoolers to their ABC's on a musical note. It begins with the last musical phrase from the familiar Alphabet Song (A-B-C-D-E-F-G . . .). Then the letter "A" appears. When your youngster types A, he or she will hear the note associated with A in the Alphabet Song. Then a second A appears on the screen, and the letter "B" is displayed on the line below it, prompting the player to type B and hear the B sound. If any other letter (or the [BREAK] key) is pressed, nothing happens. When you get to Z, the whole Alphabet Song is played, and the program starts over again at A.

This simple game allows a young child to obtain musical results from the keyboard at the same time that he or she learns the alphabet. It has given my three-year-old son Mikey a chance to get a turn at our busy Atari 800 and to gain a much-needed sense of control over its operations. He loves the game — and he's getting a lot better at his ABC's.

SYNOPSIS

This is a short educational game for young children. It runs on all Atari computers and with all memory configurations.

*Richard Seltzer writes for Digital Equipment's company newspaper, **DECWORLD**. He's written several children's books, as well as an historical novel, **The Name of Hero** (Tarcher/Houghton Mifflin), and recently had his first program published in a computer magazine.*

continued on page 18

RESTON MAKES THE ATARI CHILD'S PLAY

Reston has the books which make learning to program Atari™ computers fun.



HELLO, COMPUTER: AN INTRODUCTION TO BASIC, by Lawrence P. Huelsman, can help both teen and adult beginners learn BASIC on many computers, including Atari™, using drills, programming problems, games, cartoons and an easy conversational style.

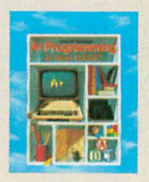


SURVIVAL ON PLANET X WITH THE ATARI™ HOME COMPUTER, by Orkin and Bogas, uses the exciting adventures of Vivian on Planet X to teach kids basic programming concepts and techniques. The fun is interspersed with short programs, illustrated by noted animator Bud Lucky.

ATARI™ LOGO ACTIVITIES, by Steve DeWitt, provides over 150 activities which encourage young and old alike to be inventive and creative when using Atari Logo™ educational language. The book includes five big projects and an in-depth discussion of Logo.™



ADVENTURES WITH THE ATARI™, by Jack Hardy, teaches you how to write adventure games in Atari PILOT™, Microsoft BASIC, and BASIC. It includes six actual adventure games to study, type in, and play, plus tips and techniques to help you create your own.



A+ PROGRAMMING IN ATARI™ BASIC, by John Reisinger, is a self-study workbook which gives you step-by-step instructions for BASIC programming on the Atari 400, 800, 600XL and 800XL™ computers. Stressing top-down programming in a fun and friendly manner, this book is perfect for school, workshop and computer camp.

If you want to make learning about Atari™ computers fun, then make Reston the teacher.



Reston Computer Group™
A Prentice-Hall Company
11480 Sunset Hills Rd.
Reston, VA 22090

Available at your local bookstore and computer retailer, or call us at (800) 336-0338.

ALPHABET MUSIC continued from page 16

```
5 REM ALPHABET MUSIC
6 REM BY RICHARD SELTZER
7 REM ANTIC MAGAZINE
10 L=64:X=2:Y=-1
20 OPEN #1,4,0,"K:"
30 GRAPHICS 2+16:GOSUB 200:RESTORE
40 L=L+1:IF L=91 THEN L=65:RESTORE
50 Y=Y+1:IF Y=11 THEN Y=0:X=X+3
60 IF X>18 THEN GRAPHICS 2+16:X=2
70 POSITION X,Y
80 ? #6;CHR$(L);
90 GET #1,K
100 POKE 16,64:POKE 53774,64
110 IF K=L THEN ? #6;CHR$(L);
120 IF K=L THEN READ A,B
130 IF K=L AND L<>87 AND L<>89 AND L<>
90 THEN SOUND 0,A,10,10:FOR I=1 TO B:N
EXT I:SOUND 0,0,0,0:GOTO 40
140 IF K=L AND L=87 THEN FOR T=1 TO 3:
SOUND 0,A,10,10:FOR I=1 TO B:NEXT I:SO
UND 0,0,0,0:NEXT T:GOTO 40
150 IF K=L AND L=89 THEN FOR T=1 TO 2:
SOUND 0,A,10,10:FOR I=1 TO B:NEXT I:SO
UND 0,0,0,0:NEXT T:GOTO 40
160 IF K=L AND L=90 THEN SOUND 0,A,10,
10:FOR I=1 TO B:NEXT I:SOUND 0,0,0,0:G
OSUB 200:GOTO 40
170 GOTO 90
180 DATA 136,100,136,100,91,100,91,100
,81,100,81,100,91,200,102,100,102,100,
108,100,108,100,121,50,121,50
190 DATA 121,50,121,50,136,200,91,100,
91,100,102,200,108,100,108,100,121,200
,91,50,102,200,108,100,121,300
200 RESTORE :FOR T=1 TO 11:READ A,B:SO
UND 0,A,10,10:FOR I=1 TO B:NEXT I:SOUN
D 0,0,0,0:NEXT T
210 FOR T=1 TO 2:SOUND 0,121,10,10:FOR
 I=1 TO 100:NEXT I:SOUND 0,0,0,0:NEXT
T
220 SOUND 0,136,10,10:FOR I=1 TO 200:N
EXT I:SOUND 0,0,0,0:RETURN
```

TALK TO YOUR ROBOT

Create motion with the Forth language

by EVAN ROSEN

Last time (**Antic**, "Has Your Robot Hugged You Today?", p. 38, January 1984), I told you how to use a BASIC listing and your Atari computer to make a servo-robot dance. If you were able to locate the required hobby servo, battery, two diodes and joystick-port connector, you may have already done so. This time, I'll discuss a roughly equivalent program in Forth and present a short sample program for each of the two languages. We'll also talk about additional axes and sensors.

SYNOPSIS

This article is the third in a series on robotics and the Atari computer. The first two parts appeared in the December 1983 and January 1984 issues of ***Antic****. The main program listing accompanying this article requires that you have the language Forth. All program listings in this article work on any Atari computer.*

Because most versions of Forth for the Atari computers are written in the fig (Forth Interest Group) dialect, our listing is in fig-Forth. Assembler mnemonics vary slightly between various Forth implementations, but the replacement of EQ with 0= should be the only change needed to accommodate your fig-Forth.

Screen 100 defines the constants PORTA and PACTL. The bits of PORTA correspond to pins 1–4 on Ports 1 and 2. PACTL, or "Port A Control," permits the control of these pins for input or output. Since we're sending pulses to a servo, we'll want to make at least one pin, pin 1 on Port 2, an output pin.

PORTSET on line 6 of screen 100 flips a bit in PACTL, programs one bit of PORTA for output, and then flips the same bit in PACTL again to finish the change.

The variables SERVO, OPULSE and TOP, defined in lines 10–12, are parameters that control the behavior of the servo. SERVO contains the one-byte rotational position of the servo. In theory, it ranges from 0 to 255, but in practice (on actual servos) you usually hit the top somewhere in the range of 180–220. The highest useful value of SERVO is contained, after calibration, in TOP. OPULSE is another calibration; it sets the zero-point of the servo's rotational travel.

The word LABEL at the end of screen 100 is defined in order to simplify the creation of the machine-code fragment DRIVER on screen 101.

FORTH'S APPROACH TO ASSEMBLY

If you're familiar with 6502 machine code, but not with Forth assemblers, the code for DRIVER should give you an idea of how Forth approaches assembly. First of all — since Forth's stack invites reverse Polish notation — things like STA PORTA become PORTA STA. Second, you'll find that most mnemonics are followed by commas (i.e., "LDX,"). Because mnemonics such as ADC or DEC also can be legitimate hexadecimal values, Forth 6502 mnemonics are marked with commas so they won't block interpretation of a hex number a programmer is trying to use. There are other approaches, of course, but this one has been used historically with the 6502.

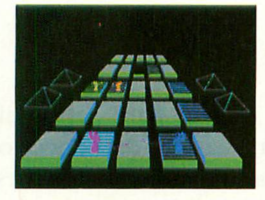
You usually won't find any explicit branch instructions or corresponding labels in Forth assembly code. Forth assemblers tend to enforce structured programming, so smart pseudo-operations such as BEGIN, WHILE, EQ, 0=, and UNTIL are used instead. In combination, these words cause the necessary branch or jump instructions to be compiled automatically.

DRIVER produces the pulses that drive your servo. Starting on line 5 of screen 101, the byte in OPULSE is loaded into the X register of the 6502. X is incremented to allow for a zero value in OPULSE. We then load the accumulator of the 6502 with 10 (hexadecimal), and store it in PORTA. This sends about five volts to pin 1 of Port 2, which begins sending pulses to the servo.

Next, a delay loop is entered. The loop consists of two NOP, or "no operation," instructions; a DEX, instruction to decrement the X register; and a test and branch (EQ UNTIL,) back to BEGIN, unless X has been decremented to zero. This

continued on page 22

SILICON WARRIOR.™
PROGRAMMED TO BLOW YOU AWAY.



The year is 2084, and the Silicon Valley is enveloped in a conflict of the fiercest magnitude. Warriors from the House of Peanut, the House of Apple, the House of Adam, and the House of Pong are competing for a computer program which will unravel the mysteries of the Universe and lead to eventual domination.

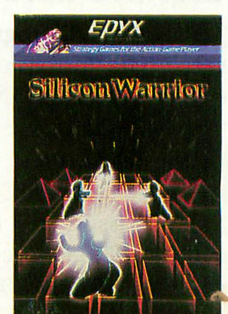
You are one of the Silicon Warriors, your mission is to program five of the chips in a row before any of your opponents can do the same. They will try to beat you to the task and steal the program or destroy you with laser fire. Or, you might even be swallowed up by a black hole "bug" in the program that appears when you least expect it. Using your joystick, you teleport from one chip to another as the battle intensifies through seven levels of play.

With the proper strategy and a dash of speed, you can be Master of the Silicon Valley.

One to four players; joystick controlled.



Strategy Games for the Action-Game Player



TALK TO YOUR ROBOT continued from page 20

loop is the "fixed length" portion of the servo pulse. (Remember that, in order to drive the servo between its endpoints, these pulses must range from a minimum duration of about 1 msec to a maximum of approximately 2 msec.)

Line 9 of screen 101 loads the value in SERVO into the X register, increments the register, and starts another delay loop. This second loop creates the "variable length" portion of the servo pulse, and thus contains positional information. That is the reason that this part of the pulse is proportional to the value in SERVO.

Finally, we load 0 into the accumulator and then store it in PORTA to send the pulse to the servo. The jump to the address at $E463 on line 14 uses the normal vertical blank deferred-exit point.

MOVING FORTH

If you have a fig-Forth system, you should be able to type in this listing, load it, and run DEMO. DEMO will make your servo oscillate, if the servo is connected according to the instructions in the January issue of **Antic**. (If you don't have the January issue, the connections were as follows: The black wire on your servo is connected to the negative side of the 6-volt lantern battery and, through a connector, to pin 8 of Port 2. The positive terminal of the battery is connected through two diodes in series, to provide about a one-volt drop, and the second diode is connected to the red (or orange) wire on the servo. The third wire on the servo is attached — through the connector — to pin 1 on Port 2.)

Try DEMO. Exit it by pressing the terminal key on your machine. The location of this key depends on which implementation of Forth you use.

If you store values in SERVO, you can control your servo directly. Set SERVO to 0, and then tweak the value of OPULSE up or down. This will cause your servo to go all the way to one stop without straining against it. Then see how high you can make SERVO before you reach the stop in the other direction, and put this value into TOP. Now, run DEMO again. The servo should oscillate through its full allowable arc at this point.

GAINING KEYBOARD CONTROL

Another quick program will give us keyboard control of the servo. The Forth program is on screen 103. Load it, and then type KEYBOARD. Next, press and hold the [+] or [*] key. This will cause the servo to move slowly in one direction or the other. Press the [X] key exit.

To use the BASIC listing, enter the BASIC code from the January issue and run it to install the machine-language routine. The new code can be added to the old code and run by GOTO 2000. Use the [+] or [*] key (as in Forth) to make your servo move.

In both versions, the slow action is created by the 10-per-second auto-repeat as you hold down the keys.

MORE AXES?

There are several ways to control more than one axis at a time. But if we stick to standard servos, we'll need at least one wire for each servo. One way to handle this is to wire a servo to each I/O pin of PORTA; this allows us to use up to eight servos — if we write some tricky software that jiggles the bits at the right time.

Another solution is to use a shift-register to take pulses from a single wire and decode them for a number of servos. This involves the use of the count-down interrupt timers on the POKEY chip; otherwise, the 6502 would spend all of its time in delay loops.

The first approach has been coded in Forth, and can control up to eight axes simultaneously. The translation to BASIC is being prepared. I'll cover one or both of these in an upcoming issue. One of the advantages of this technique, which doesn't use POKEY, is that the code can be used on other 6502 machines if a 60-hz interrupt is used. Such an interrupt can be set up on most Apple computers, for instance, with a couple of clips and a wire.

HOW *DOES* YOUR ROBOT FEEL?

If you've ever taken a joystick apart, you know that there are five switches inside: one for each of the four directions, and one for the trigger button. The joystick ports sense when these switches are closed.

By substituting other switches, you can add a sense of touch to your rudimentary robot while using very few additional parts. D-9 female connectors are available at a dollar or two apiece, and inexpensive, low-force switches can often be obtained at electronics surplus stores. You can also make your own. All you really need are two pieces of connector that touch when pressed together and separate when the pressure is released.

If you're in BASIC, you can read these switches by means of the BASIC commands STICK and STRIG. From Forth, you may want to simply byte-fetch from location 54017 for pins 1–4 on Ports 3 and 4, and from locations 53266 and 53267 for the respective triggers.

THINK, INSTEAD

But putting sensors on a robot is not the hard part at all. The real question is what to do with sensory data once you have it. So when you start to write the software for these switches, do yourself a favor: Don't write, think instead . . .

Imagine that you are blind and deaf, have no sense of direction, and lack a sense of touch except for a few points on

your body. Think about how you would have to use the sensory information you got from those few areas of contact with the outside world.

If you think there isn't very much of interest that you can do with your servo and the software tools that are currently available, you may be right. But think about the kind of software you'd need to even begin writing the ultimate robotics program you envision. Spend a week at it. Or a year. Most of the members of the "artificial intelligence" community have been at it quite a bit longer than that, but each of them had to begin somewhere. You're at the starting line now.

Evan Rosen is the co-author of Val-FORTH from Valpar International.

```
2000 POKE SERVO,TOP/2
2010 TEMP=PEEK(SERVO)
2020 K=PEEK(764):POKE 764,255
2030 IF K=6 THEN TEMP=TEMP+1
2040 IF K=7 THEN TEMP=TEMP-1
2050 IF TEMP<0 OR TEMP>TOP THEN GOTO 2
010
2060 POKE SERVO,TEMP
2070 IF K<>22 THEN GOTO 2010
```

```
Scr # 100
( Port setup and variables )
DECIMAL

54016 CONSTANT PORTA
54018 CONSTANT PACTL

: PORTSET ( -- )
  PACTL C@ DUP 4 - PACTL C!
  16 PORTA C! PACTL C! ;

128 VARIABLE SERVO
120 VARIABLE OPULSE
150 VARIABLE TOP

: LABEL 0 VARIABLE -2 ALLOT ;
                               -->
```

```
Scr # 101
( Driver routine )
HEX ASSEMBLER
100 DP C@ - ALLOT ( PAGE BNDRY )

LABEL DRIVER ( -- )
  OPULSE LDX, INX,
  10 # LDA, PORTA STA,
  BEGIN, NOP, NOP, DEX, EQ
  UNTIL, ( END FIXED LENGTH )
  SERVO LDX, INX,
  BEGIN, NOP, NOP, NOP, NOP,
   DEX, EQ
  UNTIL,  ( END VARIABLE LENGTH )
  0 # LDA, PORTA STA,
  E463 @ JMP, ( EXIT VBLANK )
                               -->
```

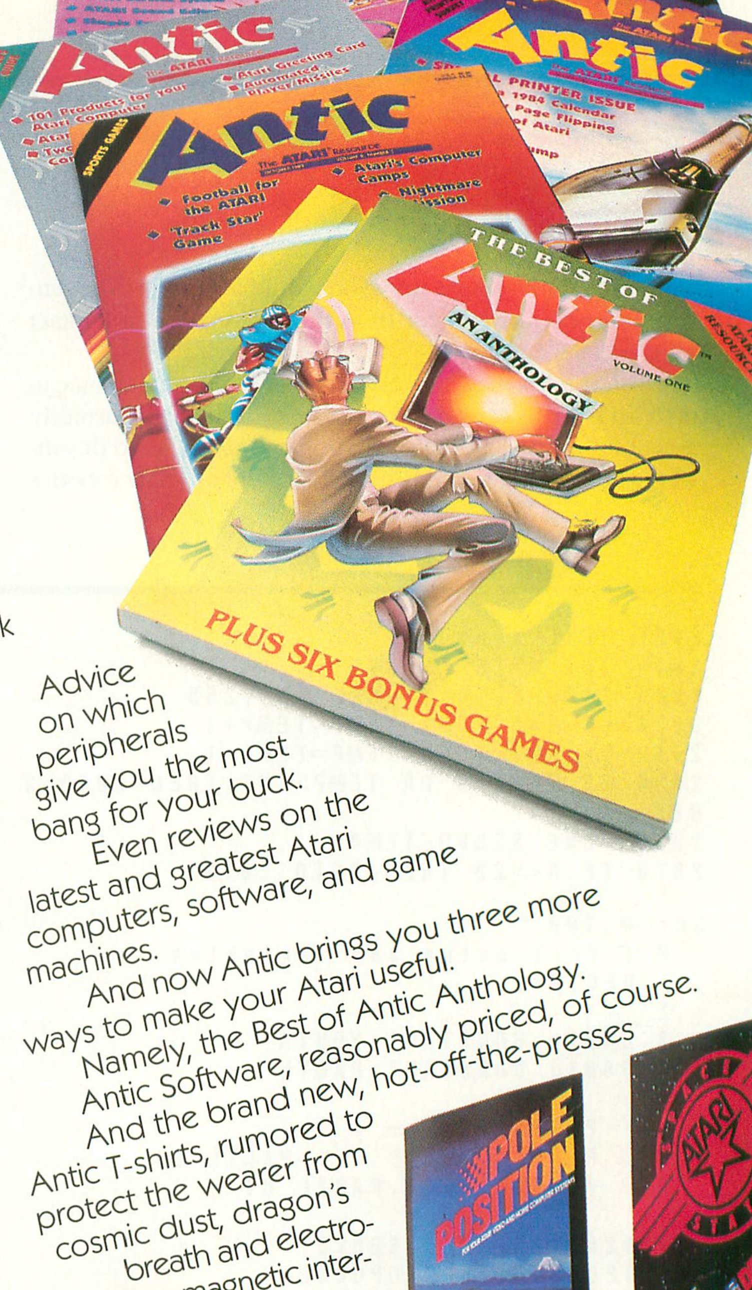
```
Scr # 102
( Oscillating Demo )
DECIMAL

: DEMO ( -- )
  PORTSET
  0 54286 C! DRIVER 548 !
  64 54286 C! ( INSTALL VBI RTN )
  BEGIN TOP @ 0
  DO I SERVO C!
   100 0 DO LOOP ( DELAY )
  LOOP
  0 TOP @
  DO I SERVO C!
   100 0 DO LOOP ( DELAY )
  -1 +LOOP ?TERMINAL
  UNTIL ;
```

```
Scr # 103
( Keyboard control demo )

: KEYBOARD ( -- )
  TOP @ 2 / SERVO !
  BEGIN
   764 C@ > R 255 764 C!
   SERVO C@
   R 6 =
   IF 1 +
   ELSE R 7 =
    IF 1 -
    ENDIF
   ENDIF 0 MAX TOP @ MIN
    SERVO !
   R > 22 =
  UNTIL ;
```





IT'S EASY. All you need is Antic, the Atari Resource.

Every month, Antic is full of things that'll help you make the most of your Atari.

Like home banking programs for keeping track of income and outgo.

Educational programs so you can keep up with the kids.

Communications programs that'll not only let you keep up with the Jones, but talk to them as well.

Plus useful utilities programs and computer languages that'll help you make your computing easier and more ingenious.

There are also games that'll make time fly.

Advice on which peripherals give you the most bang for your buck.

Even reviews on the latest and greatest Atari computers, software, and game machines.

And now Antic brings you three more ways to make your Atari useful.

Namely, the Best of Antic Anthology.

Antic Software, reasonably priced, of course.

And the brand new, hot-off-the-presses Antic T-shirts, rumored to protect the wearer from cosmic dust, dragon's breath and electro-magnetic interference...

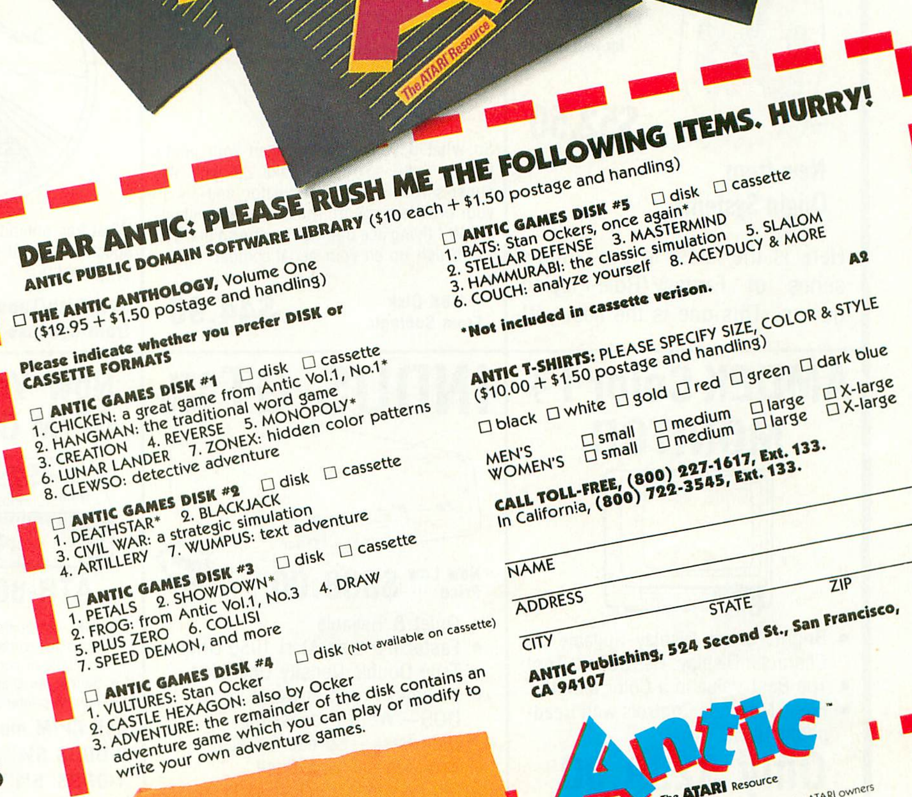


JUST THE RIGHT VOLUME

The Best of Antic Anthology covers the most popular programs and articles from our first year of publication.

Good-for-something utilities, step-by-step tutorials, plus hold-on-to-your-joystick games, including six new Atari arcade games created specially for Antic.

If you have an Atari computer, this is one book that's got you covered.



SOFTWARE AT A PRICE THAT'S NOT HARD TO TAKE

This is just part of Antic's library of public domain software, including five disks crammed-full with games like Bats, Chicken, Speed Demon, Slalom, and the ever-popular, fun-for-the-whole-family, Vultures.

You can get as many as eight of these little beauties on each disk or cassette, for as little as $10, plus shipping and handling. That $10 is not a typographical error, it really does say $10.

They couldn't be more inexpensive if you wrote them yourself.

DEAR ANTIC: PLEASE RUSH ME THE FOLLOWING ITEMS. HURRY!

ANTIC PUBLIC DOMAIN SOFTWARE LIBRARY ($10 each + $1.50 postage and handling)

☐ **THE ANTIC ANTHOLOGY,** Volume One ($12.95 + $1.50 postage and handling)

Please indicate whether you prefer DISK or CASSETTE FORMATS

☐ **ANTIC GAMES DISK #1** ☐ disk ☐ cassette
1. CHICKEN: a great game from Antic Vol.1, No.1*
2. HANGMAN: the traditional word game
3. CREATION 4. REVERSE 5. MONOPOLY*
6. LUNAR LANDER 7. ZONEX: hidden color patterns
8. CLEWSO: detective adventure

☐ **ANTIC GAMES DISK #2** ☐ disk ☐ cassette
1. DEATHSTAR* 2. BLACKJACK
3. CIVIL WAR: a strategic simulation
4. ARTILLERY 7. WUMPUS: text adventure

☐ **ANTIC GAMES DISK #3** ☐ disk ☐ cassette
1. PETALS 2. SHOWDOWN* 4. DRAW
2. FROG: from Antic Vol.1, No.3
5. PLUS ZERO 6. COLLISI
7. SPEED DEMON, and more

☐ **ANTIC GAMES DISK #4** ☐ disk (Not available on cassette)
1. VULTURES: Stan Ocker
2. CASTLE HEXAGON: also by Ocker
3. ADVENTURE: the remainder of the disk contains an adventure game which you can play or modify to write your own adventure games.

☐ **ANTIC GAMES DISK #5** ☐ disk ☐ cassette
1. BATS: Stan Ockers, once again*
2. STELLAR DEFENSE 3. MASTERMIND
3. HAMMURABI: the classic simulation 5. SLALOM
6. COUCH: analyze yourself 8. ACEYDUCY & MORE

A2

***Not included in cassette verison.**

ANTIC T-SHIRTS: PLEASE SPECIFY SIZE, COLOR & STYLE ($10.00 + $1.50 postage and handling)

☐ black ☐ white ☐ gold ☐ red ☐ green ☐ dark blue

MEN'S ☐ small ☐ medium ☐ large ☐ X-large
WOMEN'S ☐ small ☐ medium ☐ large ☐ X-large

CALL TOLL-FREE, (800) 227-1617, Ext. 133.
In California, **(800) 722-3545, Ext. 133.**

NAME

ADDRESS

CITY STATE ZIP

ANTIC Publishing, 524 Second St., San Francisco, CA 94107



ANTIC™ is an independent monthly publication for ATARI owners and users. ATARI™ is a registered trademark of Atari, Inc.

SHOW YOUR TRUE COLORS

And there's no more colorful way to do that than with an Antic T-shirt.

They're available in a variety of adult sizes and colors, some of which are even found in nature.

They're made of a comfortable cotton and with the Antic logo emblazoned across the front, they immediately identify you as someone in possession of superior taste, intelligence and breeding.

They also look nice under a sweater.



ANATOMY OF AN 800XL

Inside the belly of the beast

by ROBERT DeWITT
Managing Editor

For those of you who are curious about the innards of the XL computers, here is a guided tour of the inside of an Atari 800XL. We were treated to this tour by the engineering staff at Atari. Please note that there are some differences between the 800 XL and the other XL computers.

It isn't hard to take the computer apart, but if you open the case you void your warranty. Perhaps this article will quench your desire to do so.

The top of the case is attached by screws to the bottom, and is easily removed, but you should be careful with the ribbon connector that plugs the keyboard into the printed-circuit board. This connection is the most delicate in the whole assembly, and frequent detachment of it is not recommended.

The metal plate that covers most of the printed-circuit (PC) board suppresses radio-frequency (RF) interference. The exposed arrays of resistors and condensers at the immediate right of the plate also serve that function. The controller ports are mounted at the extreme right, and the serial I/O plug is at the upper right.

Under the RF plate are the other components. Arranged along the front are the major chips. From left to right they are: GTIA, ANTIC, the 6502, and PIA. POKEY is directly above PIA. In the 600XL, the positions of GTIA and ANTIC are reversed.

The two chips just to the left of the keyboard connector are the keyboard decoders. Above them are the Operating System and BASIC chips.

continued on next page

The cartridge slot is obvious. The parallel bus is located just above the cartridge slot at the back of the computer.

Between the cartridge slot and the 6502 are the address-decoder chips, the RAM-timing chips, and several chips that are described as "glue" for the system. There are far fewer glue chips in the XLs than in previous Atari computers.

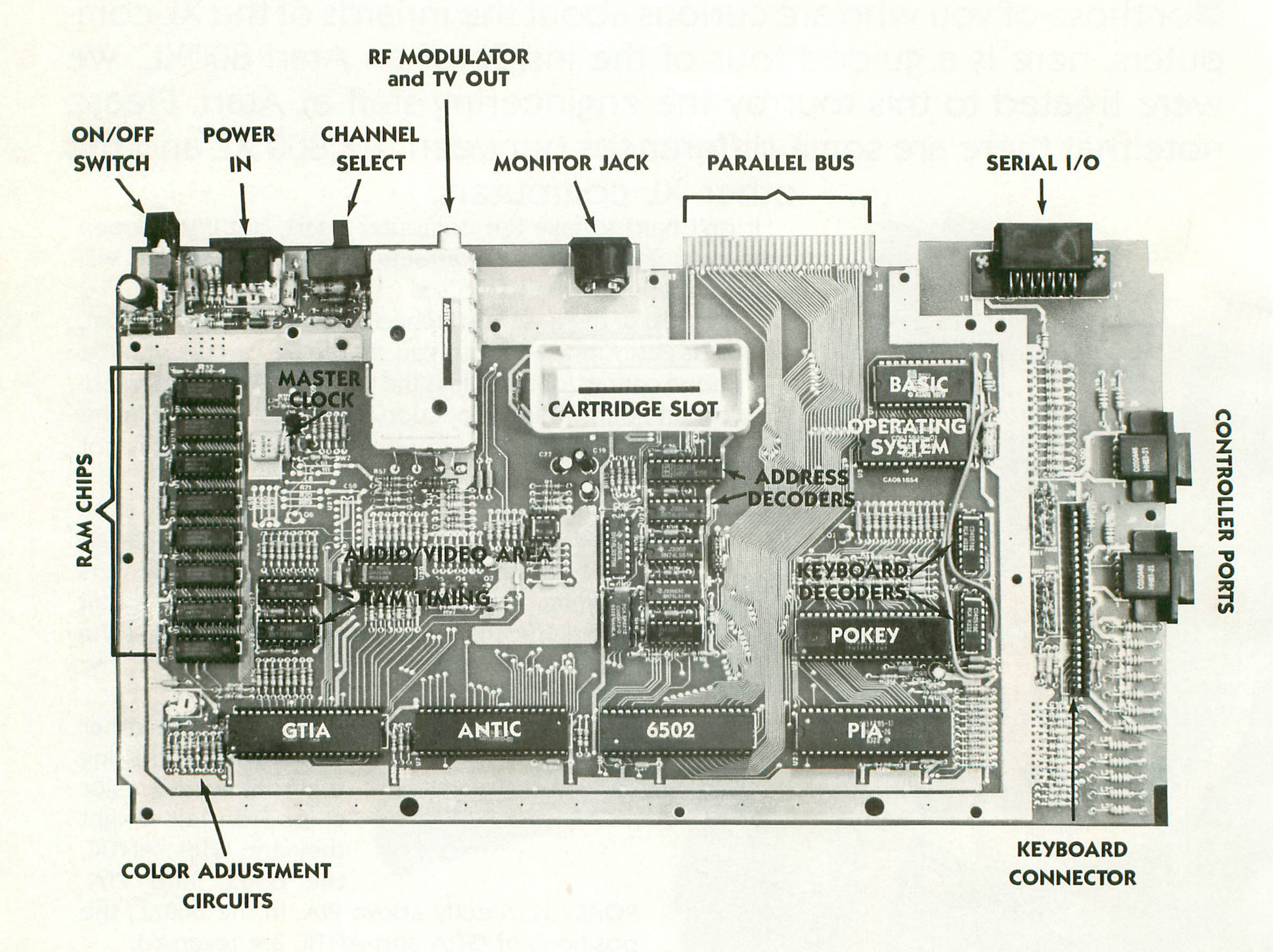
Immediately above the ANTIC chip is the audio/video analog circuitry that culminates in the rectangular RF modulator box at the back of the machine. To its left is the monitor jack. Note that this jack has only three leads: audio, luminance, and composite video; chroma is not independently available as it was with the Atari 800.

To the right of the RF modulator are the channel switch, the power-in connector, and the ON/OFF switch. Note that the power connector is a DIN plug. It is designed to prevent the accidental connection of a power cord from older Atari computers.

The eight chips that line the left side are "64K by one-bit" RAM chips. In the front left corner is the color-adjustment circuit.

Above the GTIA and to the right of the RAM chips are the dynamic-RAM address multiplexors. Further up is the master clock. The intervening gap is reserved for the circuits needed by the XLs prepared for international markets.

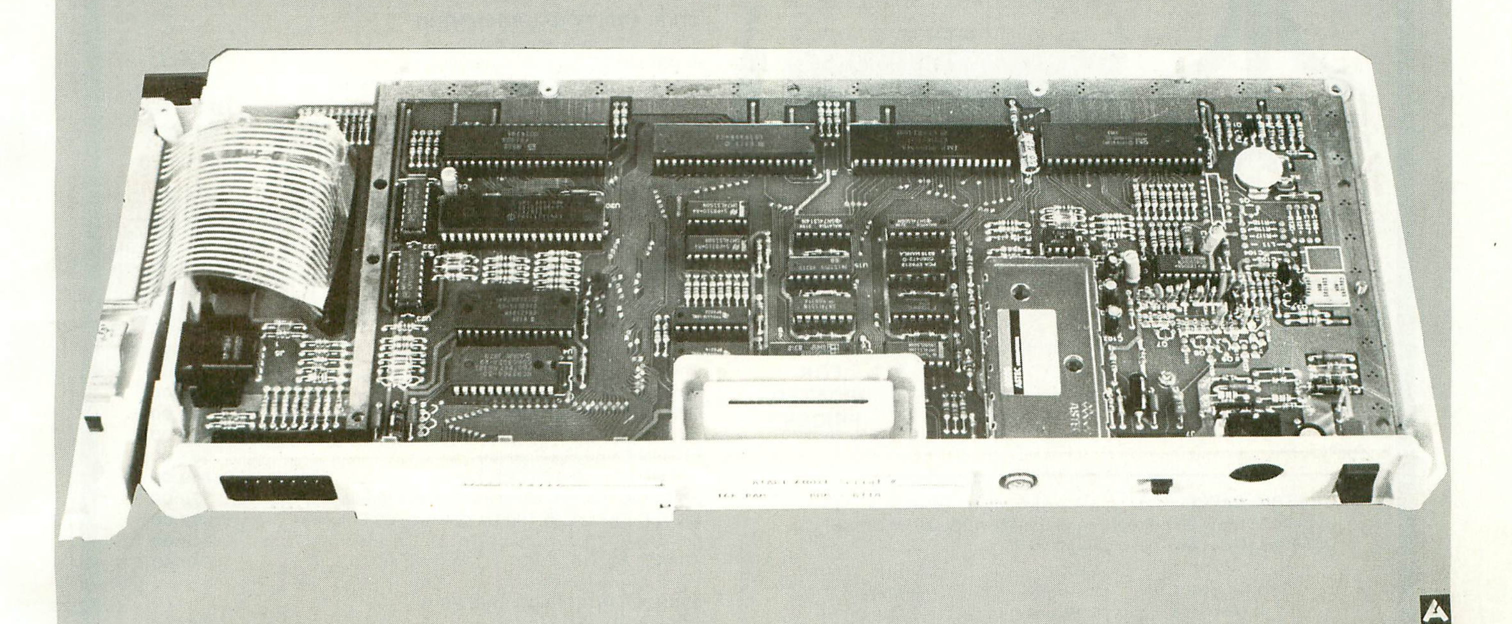
When you power-up the 800XL, the first thing that starts is the clock. The reset stabilizer then initializes certain latches in the PIA, so that the 6502 can find the Operating System, and sets ANTIC to a known state of non-display. Then the 6502 kicks in, and goes to address $FFFC. This vectors it to the initialization routines.



800XL



Rear view of the 600XL



Always the Lowest Prices on the Finest Quality at. . . COMPUTER CREATIONS





ATARI HOME COMPUTERS

ATARI 600™ Home Computer (16K RAM)	CALL FOR UPDATED PRICES
ATARI 800XL™ Home Computer (64K RAM)	

DISK DRIVES

RANA 1000	CALL FOR UPDATED PRICES
TRAK AT D2 (Single/Double Density/parallel int./4k buffer)	
TRAK ATD4 (Dual Drive)	
TRAK TURBO DOS	
INDUS GT (Free Software)	
ASTRA 1620	
ATR - 8000	

GENERIC DISKS

Generic 100% Defect-Free/Guaranteed MINI-FLOPPY DISKS
Diskettes (1 Box Min.) - 10 per box

	SS/DD	DD/DD
1 or 2 Boxes	17.49/box	20.99/box
3 - 9 Boxes	15.99/box	19.99/box
10+ Boxes	14.99/box	18.99/box

Bulk Diskettes with Sleeves - Price per Disk

	SS/DD	DD/DD
10 - 29	1.59	1.99
30 - 99	1.49	1.89
100+	1.45	1.79

HARDWARE COVERS

ATARI 400	5.95
ATARI 800	5.95
ATARI 1010	5.95
ATARI 410	5.95
ATARI 810	5.95
ATARI 1200	5.95
ATARI 1025	5.95
ATARI 600XL	5.95
ATARI 1050	5.95
ATARI 800XL	5.95
ATARI 1027	5.95
EPSON MX-80	5.95
EPSON FX-80	5.95
EPSON RX-80	5.95
EPSON RX-80 FT	5.95
EPSON FX-100	7.49
GEMINI 10X	5.95
GEMINI 15X	7.49
RADIX 10	5.95
RADIX 15	7.49
POWERTYPE	5.95
DELTA-15	7.49
RANA 1000	5.95
INDUS	5.95
TRAK	5.95

commodore

COMMODORE 64 HARDWARE

1530 Datasette	65.00
Maxiset	56.00
1541 Disk Drive	239.00
1650 Modem	79.00
1526 Printer	269.00

COMMODORE INTERFACE CABLES

Cardco A	54.00
Cardco B	39.00
Cardco GT	69.00
Tymac	79.00

MODEMS

HES Modem I	60.00
Transterm Software	29.00

ACCESSORIES

Data 20 (80 CHR.)	129.00
Data 20 (80 CHR./CPM)	229.00
Koala Pad	69.00

ATARI ADDITIONAL EQUIPMENT

ATARI 850™ Interface Module	CALL FOR UPDATED PRICES
ATARI 1010™ Program Recorder	
ATARI 1020™ 40-Column Color Printer/Plotter	
ATARI 1025™ 80-Column Printer	
ATARI 1027™ Letter Quality Printer	
ATARI 1030™ Direct Connect Modem	
ATARI 1050™ Disk Drive	
CX77 ATARI Touch Tablet™	

PRINTERS

GEMINI 10X (80 Column)	279.00
GEMINI 15X (136 Column)	399.00
DELTA 10 (80 Column)	479.00
RADIX 10 (80 Column)	639.00
RADIX 15 (136 Column)	749.00
Powertype Daisywheel	379.00
Epson RX-80 (80 Column)	339.00
EPSON RX-80 (80 Column)	339.00
EPSON RX-80 FT (80 Column)	469.00
EPSON FX-80 (80 Column)	555.00
EPSON FX-100 (136 Column)	749.00
AXIOM AT-100 (80 Column) Includes Interface & Cable	229.00
SILVER-REED EXP 500 Daisywheel (80 Column)	399.00
OKIDATA 92P	449.00
TRANSTAR 315	549.00

PRINTER INTERFACE CABLES

APE FACE	65.00
ATARI Universal Cable	59.00
MPP-1150 Parallel Printer Interface	79.00
Parallel Printer Cable	30.00

PRINTER RIBBONS

Gemini Printers (Black/Blue/Red/Purple)	3.00
Epson Printers	6.00

MONITORS

Gorilla™ Hi Res 12" Non-Glare Screen	89.00
Gorilla™ Hi Res 12" Non-Glare Amber Screen	99.00
Sakata SC 100 Color Screen	239.00
NEC 1260	109.95
Monitor Cable	10.00

MODEMS

MPP-1000C Modem	129.00
Signalman Mark II Modem	79.00
Hayes Stack Smartmodem (300 BAUD)	239.00
Hayes Stack Smartmodem (1200 BAUD)	549.00
Novation J CAT	119.00
Novation 103 Smart CAT	189.00
Novation Auto CAT	199.00

DISKETTE/ CARTRIDGE/ CASSETTE FILES

Flip 'N' File	$3.95
Flip 'N' File 15	6.95
Flip 'N' File 50	26.95
Flip 'N' File/The Original	19.95
Cassette 'N' Game File	14.95
Disk Bank	3.99
Disk Bank (Plastic)	1.99
Blank 30 Min. Cassettes	1.29
Blank 10 Min. Cassettes	.99

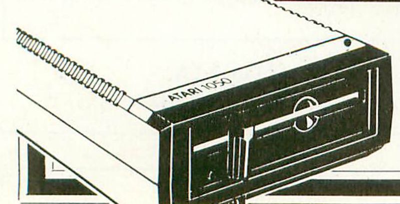
RAM (MEMORY) BOARDS

Intec 64K Board (400)	109.00
Intec 48k Board (400)	89.00
Intec 32k Board (400)	39.00
Microbits 64k (600XL) Expansion	129.00

80 COLUMN BOARDS

BIT 3	249.00

To order call TOLL FREE
1-800-824-7506





COMPUTER CREATIONS, Inc.
P.O. Box 292467 - Dayton, Ohio 45429
For information call: (513) 294-2002 (Or to order in Ohio)

Master Card - Visa - C.O.D. (Add $2.50). All orders add $3.50 Shipping and Handling
Ohio Residents add 6% for Sales Tax. **ASK FOR OUR FREE CATALOG**



EXPLORING THE XL

one programmer's perspective

by MATTHEW RATCLIFF





As an avid user of Atari computers who has been programming on an Atari 400 for over three years, I've been interested in the new XL line for a long time. So, two weeks ago, I bought an Atari 800XL. This article combines information I've gained from my own experiences with the machine and information I've culled from various Atari sources in recent weeks. The opinions expressed here are my own.

Many people, noting that the 800XL is advertised as a 64K machine, assume that it gives you 16K more RAM (random-access memory) for programming than the 48K Atari 800. However, the 800XL actually has about 20 fewer bytes of free RAM in BASIC than the 800 does. Apparently, Atari intended to "bank select" (swap ROM for RAM) the extra RAM in certain programming environments. As a result, they added some "pointers" in lower RAM to help the Operating System (OS) keep track of which RAM/ROM configuration was in effect. These pointers are largely responsible for using up the extra bytes of RAM noted above.

SYNOPSIS

This article looks at a number of interesting aspects of Atari's new line of XL computers. If you have an XL machine and Synassembler, the accompanying program listing will convert Synassembler so that it can run on your XL machine without the Translator program.

The 600XL and 800XL also come with "built-in BASIC." This is a revision (Rev. B) of the earlier, cartridge-based Atari BASIC (Rev. A). Built-in BASIC corrects certain bugs in Rev. A, but unfortunately, it introduces two new ones. A Rev. C BASIC has been prepared to correct these bugs, and it is now available (see below). Meanwhile, the bugs in Rev. B are as follows:

- Each time you SAVE and LOAD a BASIC program, the file is expanded by 16 bytes. This is cumulative, so in program



development you may run out of memory sooner than you should, especially if you have only 16K RAM.

• Your keyboard may lock up unpredictably while you are programming, or you may get garbage in the list. These two flaws are due to the same bug, and the occurrence is fatal to that run of the program.

I've used my 800XL to edit large (over 20K) BASIC programs for two weeks now without experiencing the keyboard lock-up bug. This was a relief, but there were some problems I did encounter that haven't been confirmed by Atari. These problems appear only intermittently, but when they do occur, they're trouble.

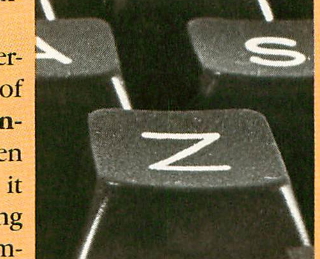
After a long session with a BASIC program, I always LIST it to disk, type NEW, and then ENTER the program, in order to clean up the variable-name table. On several occasions, with the 800XL, I've encountered ERROR 9 (string not dimensioned) when I ran the cleaned-up program. Using the CLR command doesn't help in this case, nor does pressing [SYSTEM RESET]. If I subsequently SAVE and LOAD the program, it runs fine, but I'm still not able to ENTER and successfully RUN the LISTed version. This seems to occur only with large programs.

Another LIST and ENTER problem occurs only with some files. The ENTERed program refuses to run. If you have wisely SAVEd the file as well, LOAD it and delete one byte from a REM statement or from some other harmless place. Now you should be able to LIST and ENTER it without difficulty.

I've used the Editor of my Assembler Editor cartridge (ASM/ED) to edit LISTed BASIC programs for three years now without incident, because ASM/ED's FIND and REPLACE commands make it easy to clean up a program listing. I've had problems ENTERing a LISTed BASIC file from the 800XL, however. I frequently get ERROR 137 (truncated record). I can't figure this out, since BASIC ENTERs the same file with no problem.

NEW AND IMPROVED OPERATING SYSTEM?

The 600XL and 800XL Operating System (OS) is very similar to that of the 1200XL. As a result, it suffers from a similar lack of compatibility with much third-party software. Fortunately, Atari has made the Translator program available on disk for the XL line. If you load the Translator, you can load and run any third-party software. However, if you press [SYSTEM RESET] with Translator in memory, this will probably cause your system to crash, forcing you to reload (unless you use the accompanying program as explained below).



Although it's no longer commercially available, my assembler of choice is Synapse Software's **Synassembler**. Unfortunately, when I first tried it with the XL OS, it didn't work properly. After a long session of disassembly and comparison of the old and new Operating Systems, I found out why. Several commonly-used nonstandard routines have new locations in 800XL OS. Among these are the only two illegal entry points used by Synassembler: the "put character" routine (EOUTCH), and the "get character" routine (EGETCH). The EOUTCH routine has been moved from $F6A4 to $F2B0 in the 800XL, and the EGETCH routine has been moved from $F63E to $F24A.

The BASIC program listing at the end of this article reads any unprotected machine-language file from disk and replaces all JSR instructions to the old OS EOUTCH and EGETCH routines with the new XL entry points. When I used this program on Synassembler, it ran perfectly on the 800XL without the Translator program, and recovered properly from a press of [SYSTEM RESET]. If your program uses only the two invalid calls noted above, this BASIC program should do the trick for you. Note that it also converts data within the program that happens to look like the invalid JSR's. Any program that this routine successfully converts should no longer require the annoying "double boot" procedure (with Translator) noted previously, and should operate correctly even if [SYSTEM RESET] has been pressed. But before altering your program, make sure that you've made a backup copy, in case the conversion is unsuccessful.

DOS 3

Most of the Atari 1050 disk drives sold so far include DOS 2.0S, but Atari will send a copy of DOS 3 free of charge to



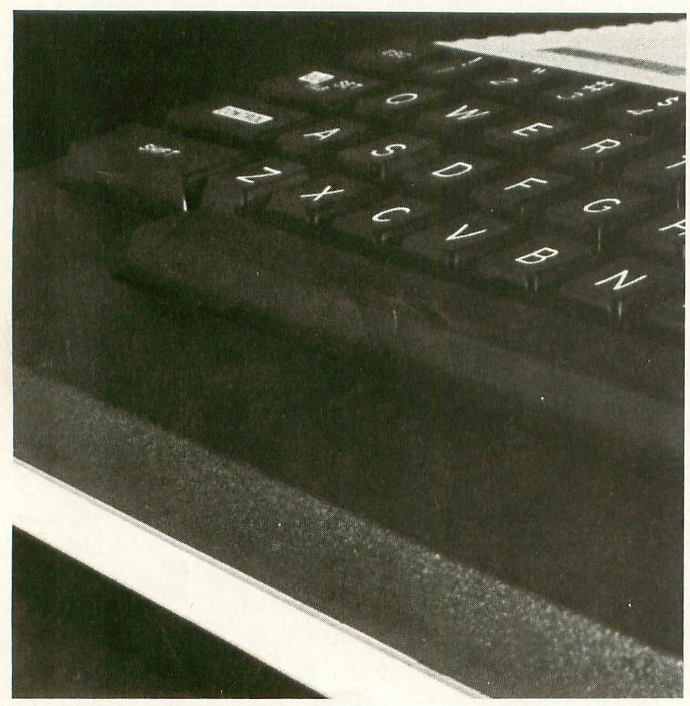
owners who write or call Atari for it. Be sure to give them the serial number of your 1050 drive. DOS 3 increases the

continued on next page

the capacity of a disk from 88 Kbytes to 127 Kbytes. (With true double-density storage, a disk can hold approximately 180 Kbytes.) The "sectors" of DOS 2.0S have been supplanted by "blocks" in DOS 3. Sectors hold 128 bytes each; blocks store 1000 bytes. DOS 2.0S gives you 707 sectors per disk, while DOS 3 gives you 128 blocks.

This can be a problem if you use small files. DOS 3's basic unit of storage, the block, takes up 1000 bytes. If your file occupies 1001 bytes, it will require two blocks of disk space, thus wasting 999 bytes! In a similar situation with DOS 2.0S, only 127 bytes are wasted. As a result, the potential additional storage space that DOS 3 offers can easily be lost if you tend to use a lot of short files.

I can only speculate that this system may have been set up in anticipation of the double-sided disk drive originally promised for the 1450XLD. If Atari's programmers allowed for a one-byte (0–255) pointer to any block on the disk, then any block reference greater than 127 refers to side two of the disk.



BETTER THINGS TO COME

Despite the minor problems and restrictions I've noted here, the new XL machines possess considerably more potential than did the 400/800 line. The built-in expansion port on the XLs makes possible a host of add-ons that are similar to those currently available for the Apple. You also can expect to see inexpensive 80-column boards, additional bank-select RAM cards, and possible DMA (direct-memory-access, or high-speed) hard disk drives. Such features can turn your XL into a very serious computing machine.

It seems that the hardware is always out the door before the documentation is complete. There is no mention in the manuals accompanying the XL machines of how to make use of the extra RAM these machines offer. However, with Translator installed, you can make use of the 4-Kbyte block of RAM from $C000 to $CFFF (49152–53247) from either BASIC or ASM/ED. Your favorite machine-language utility routines can be safely stored here (but you should first either make sure that they're relocatable, or reassemble them at the new address). You can also use this space to store character sets and Player/Missile data. And, by using certain programming tricks, you also can store strings (and possibly variables) here, thus giving yourself — in effect — four additional Kbytes of programming space.

You can order the Translator disk directly from Atari, if there isn't an Atari dealer near you. Simply send your name, address, the serial number of your XL computer, and a check for $9.95 to: Atari Customer Relations, 1312 Crossman Ave., P.O. Box 61657, Sunnyvale, CA 94088.

Those of you who are more technically oriented should order an XL addendum to the Atari Technical Notes (for $4.95) from the same address. If you want the Technical Notes as well, add $29.95. California residents should add 6 percent state sales tax.

RANDOM THOUGHTS

The power supply of the XL machines is completely external to the system. As a result, the XLs are smaller than the 400/800 computers. However, the new power supply is about twice as large as the old one, and can get quite warm to the touch. Keep it in a place where air can circulate around it freely, and the supply will run cooler and last longer.

The XL keyboard has a rather "sticky" feel, and the keys "rock" on their supports and give somewhat less than desirable feedback to the typist. I find that the 800XL keyboard lacks the "springy" touch that is characteristic of the 800 keyboard.

ATARI BASIC'S REVISION C

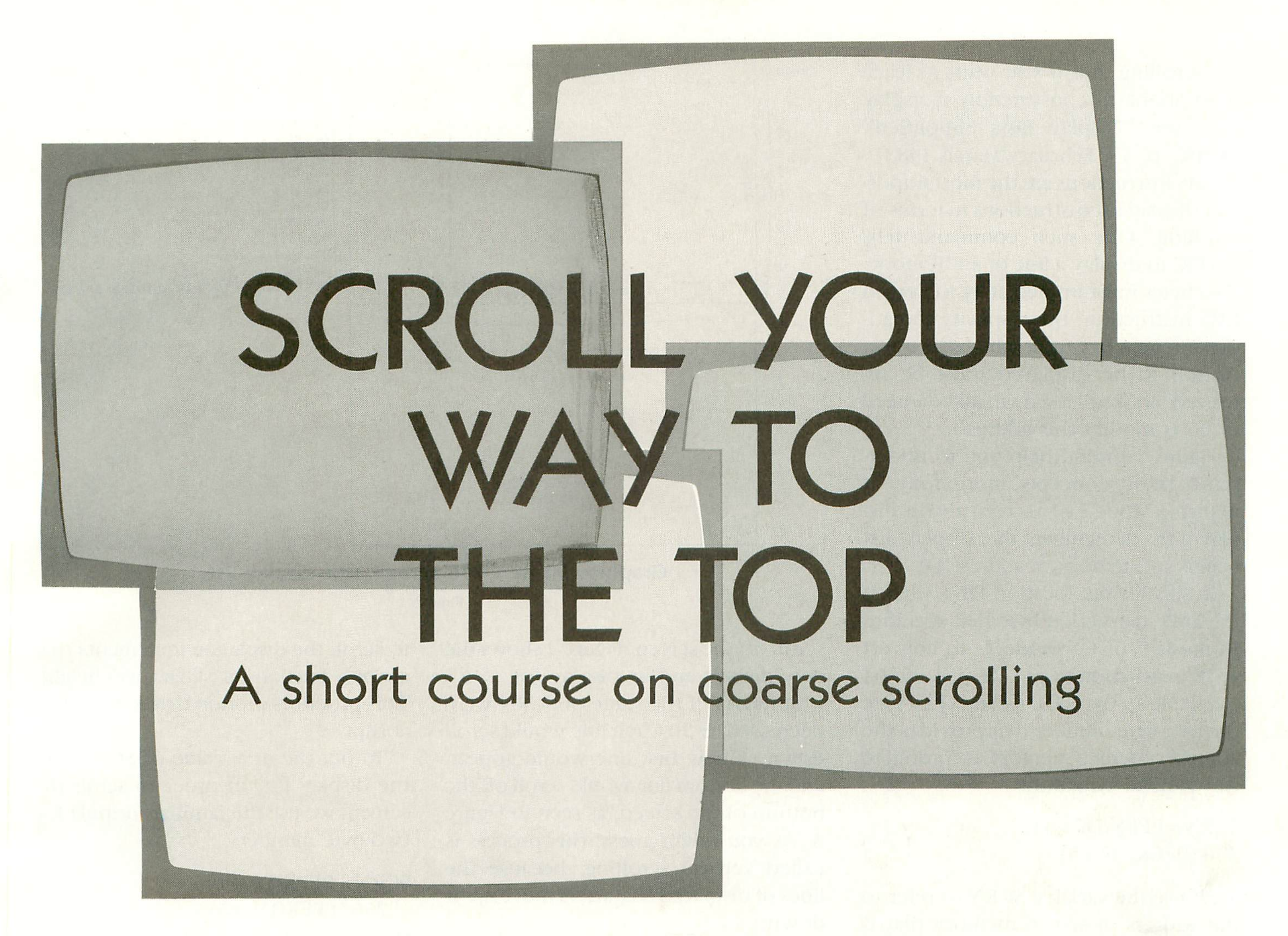
By the time you read this, a new, fully debugged version of Atari BASIC will be available. Called "Revision C," the cartridge will be sold for $15 on a first-come, first-served basis by Atari Customer Relations at the above address.

You can tell which revision of BASIC you have by PEEKing location 43234. The byte at this location in Revision A (the version included in all cartridges to date) is 162; in Revision B (the version built into all 600XL and 800XL computers to date), it's 96. If you PEEK this location in Revision C, you'll get the value 234.

Matthew Ratcliff is an electrical engineer and a microcomputer enthusiast. He owns a customized Atari 400 with 48K, and has been programming in BASIC for six years.

continued on page 68





SCROLL YOUR WAY TO THE TOP

A short course on coarse scrolling

by CHRIS CHABRIS

Scrolling is one of the easiest graphics techniques to implement on the Atari home computers. Reduced to its simplest form, it requires only one or two POKE commands. But scrolling is also the most difficult graphics effect to master. Coarse scrolling can be produced in BASIC, and is useful for most applications. I'll only touch on fine scrolling, which is beyond the scope of this article.

Before we delve into scrolling itself, we'll review the Atari display list. If you're familiar with the creation of custom display lists, however, you may want to skip to "Starting to Scroll."

The display list is a program for the Atari computer's ANTIC chip, which controls the display of text and graphics on the screen. It resides in RAM along with any user program, and is written by the Operating System (OS) whenever

SYNOPSIS

This article serves as an introduction to the graphic animation technique called scrolling. The demonstration program requires BASIC and a minimum of 16K RAM, and runs on all Atari computers.

you use a GRAPHICS command.

The display list is always located in memory at the address given by:

PEEK(560)+256*PEEK(561)

Memory locations 560–561, which are labeled SDLSTL, contain a two-byte address that ANTIC uses to find its program. We'll use the variable DL to represent the address stored in these two bytes.

Display lists consist of four basic types of instructions: Blank, Jump, Instruction Register (IR), and Load Memory Scan (LMS). A Blank instruction tells ANTIC to leave from one to eight scan lines blank. There are two different Jump instructions. The first tells ANTIC to continue displaying data, but to display it from a different area of memory. The second signals a return to the beginning of the display list. This permits ANTIC to "draw" the screen that will follow the next vertical-blank period.

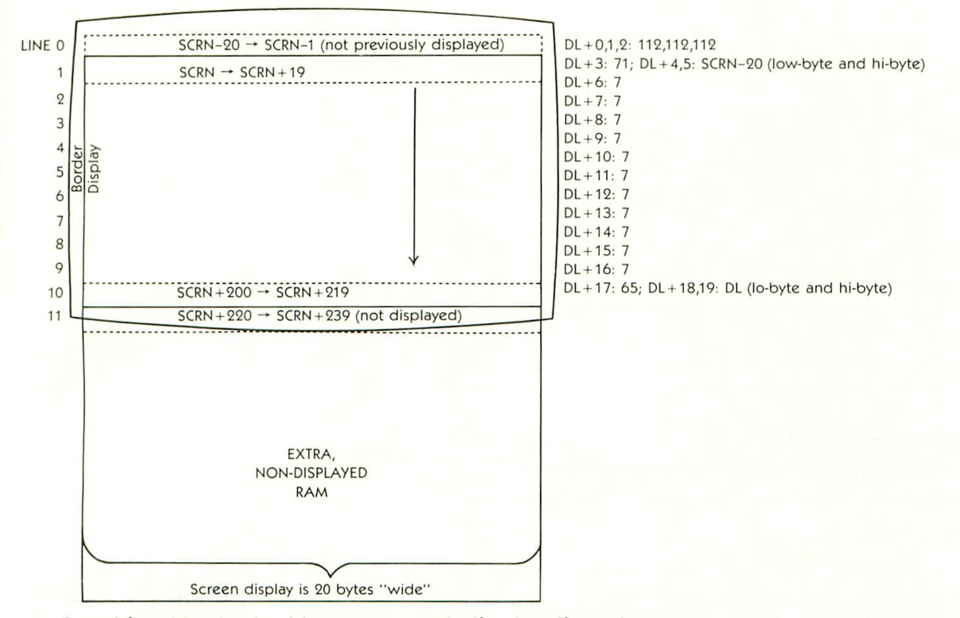
IR instructions range in value from two to 15 (decimal). They tell ANTIC to display one line of the corresponding mode. Note that IR, or ANTIC mode numbers, do not correspond to the OS Graphics Mode numbers. As Table 1 shows, custom display lists give you access to five modes that aren't available through the BASIC GRAPHICS statement. IR commands are not important

continued on next page

for scrolling, but if you want to learn

LINE 0

SCRN → SCRN+



Graphics Mode 2+16 screen and display list after downward scrolling.

Figure 3

byte address that is stored in a variable and break it into two variables. These are then POKEd back into memory. This is not the most efficient method in BASIC, but it will do for now.

HORIZONTAL SCROLLING

Let's return to our original display (Figure 1), and see what happens if we increase SCRN by one. The answer is that all of the characters are pushed one space to the left. But this does not constitute horizontal scrolling, because it causes the leftmost character on lines 1 through 11 to move to the right end of the line above it, and also causes a character from non-displayed RAM to sneak into the lower-right-hand corner of the display at the end of line 11. As a result, this method does not effect true horizontal scrolling.

The natural arrangement of screen memory, as shown in Figure 1, is responsible for our problem here. In this arrangement, a number of 20-character lines, or 20-byte blocks of memory, are stacked one on top of the other. This is a logical arrangement and is well suited to vertical scrolling, but a variation is required for horizontal scrolling. This configuration is shown in Figure 4.

Note that each line in Figure 4 is forty characters long — or twice the length of a normal Graphics Mode 2+16 line. This creates a total display area that is twice as large as the normal screen memory. Consequently, we can now scroll left and right across the memory area.

But to do so, we need to use the rewritten display list that accompanies Figure 4. It includes an LMS instruction for each of the twelve lines, and increases the address following each instruction by forty bytes. When ANTIC reads this display list, it begins each 20-byte displayed line 40 bytes ahead of the previous one (in memory). The result is a display of the left half of the total memory area.

To scroll the screen to the right or left, simply loop through the display list, adding or subtracting one to or from each address. For example:

```
FOR L=DL+4 TO DL+37 STEP 3
MEM=PEEK(L)+256*PEEK(L+1)
MEM=MEM+1:REM
  Or "MEM=MEM-1"
MEMH=INT(MEM/256)
MEML=MEM-MEMH*256
POKE L,MEML:POKE L+1,MEMH
NEXT L
```

All that this really adds to the previous program fragment are the looping commands in the first and last lines. Otherwise, we're still basically modifying an address in memory.

As you can see, scrolling itself is not very difficult. Managing display memory (allocating space properly) and creating interesting displays (filling that space) are the aspects that make scrolling a demanding technique. Now, let's see how all of these elements can be combined effectively in a complete program.

USING THE PROGRAMS

Type Listings 1 and 2 into your computer and verify your work with the TYPO program (**Antic**, p. 42, February 1984). RUN Listing 2 first to produce the disk files that Listing 1 will access. If you get a "BAD DATA" message, fix the guilty line and reRUN the program. Repeat the process until you see the "FILES GENERATED" message.

When RUN, Listing 1 will blank the screen for a few seconds while it creates a few machine-language subroutines, loads a character set and map file from disk, and writes the display list from Figure 4. Next, the upper-left-hand corner of a 24-line by 40-character map will appear. Using a joystick plugged into Port 1, you can scroll the map in all eight directions. As you can see, it moves pretty quickly.

continued on next page

TABLE ONE

ANTIC Instructions

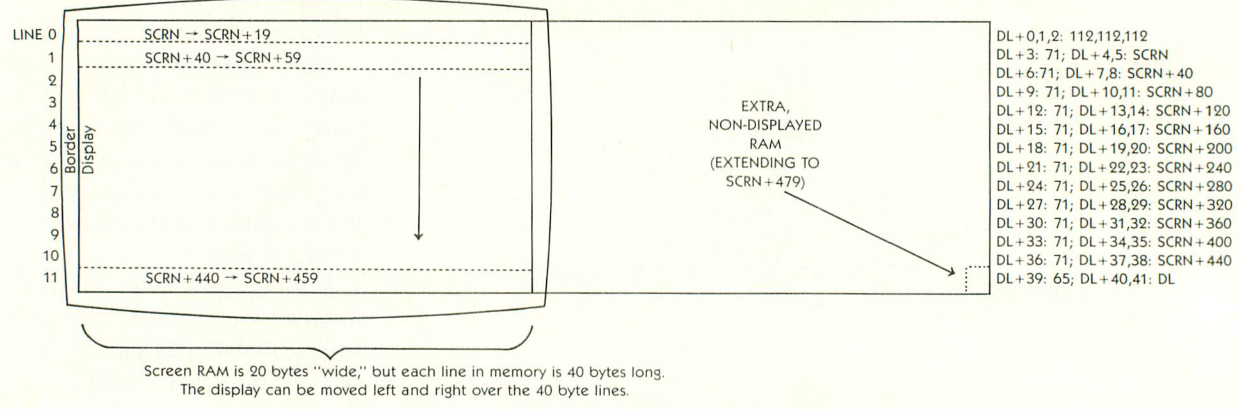
BASIC mode number	IR instruction	LMS instruction
0	2	66
–	3	67
–	4	68
–	5	69
1	6	70
2	7	71
3	8	72
4	9	73
5	10	74
6	11	75
–	12	76
7	13	77
–	14	78
8–11	15	79

NOTES:

A) Multiples of 16 (from 0 to 112) cause 1 to 8 scan lines to appear in the background color.

B) To cause a display list interrupt (DLI), add 128 to the ANTIC instruction (IR or LMS).

C) GTIA modes (9,10,11) all use IR instruction 15 and are selected by using location 623 (GPRIOR). For more information, see **Antic**, "Window on GTIA," p. 48, April 1983.



Graphics Mode 2+16 screen and display list configured for horizontal scrolling.

Figure 4

HOW THE PROGRAM WORKS

See Figure 5 for a diagram of the map in memory. The screen initially displays the characters in the second quadrant of the map, but it can be "moved" to show any 12×20 block.

In line 1600, POKE 559,0 turns off the ANTIC chip; as a result, nothing is displayed on the screen. This increases processing speed by at least 25 percent. Lines 1800 to 4000 create three machine-language subroutines; the assembly-language source code for these is given in Listing 3. More on them later.

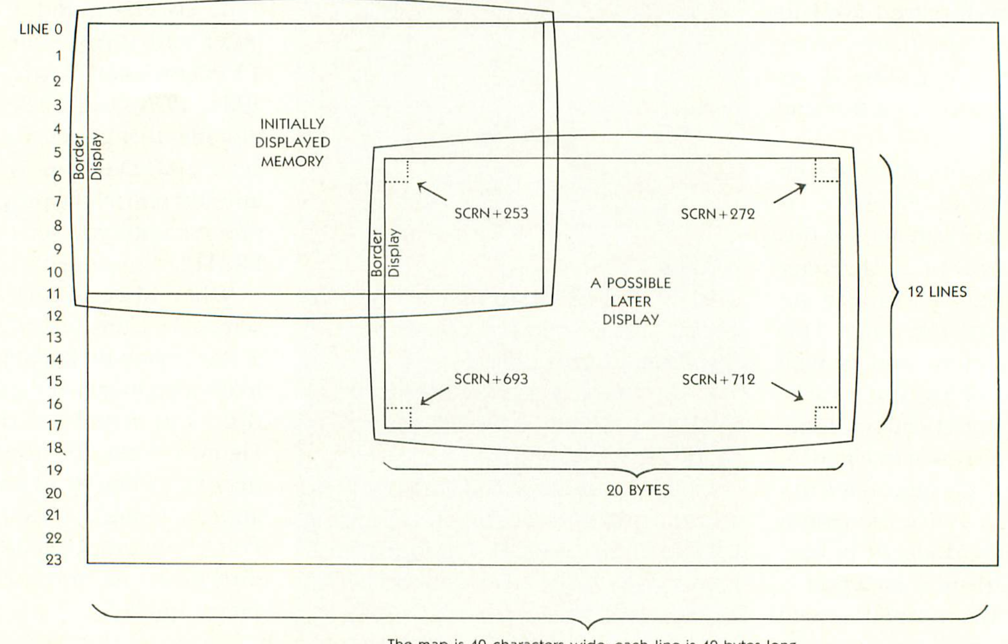
Line 4200 is very important. It first finds the number of 256-byte "pages" of memory available and reserves twelve of them (3K) for our own use. Map screen data is loaded into an address 3K below the top of memory; the character set is loaded into an address 2K above that. If you POKE the map address minus one back into location 106, this tells the OS not to disturb the safe 3K while it is being used. POKE 756,CP tells ANTIC where to find our redefined character set of map symbols.

Lines 4300–4400 use the machine-language subroutine in DF$ to load the character set and map files into memory at the defined addresses. This subroutine can load or save data to or from any area of RAM faster than BASIC can; see Listing 3 for details on using it in your own programs. Note that the file must be OPENed and CLOSEd from BASIC. Line 4500 POKEs the appropriate color-luminance values into the five color registers used in Graphics Mode 2.

Lines 4700–5000 build a display list by taking the value of MAP (equivalent to SCRN from above) and increasing it by 40 in a loop for each successive line. Before the 12 LMS-address pairs, the three 112's cause 24 blank scan lines to be displayed at the top of the screen. Then the bytes 65,0,6 tell ANTIC to start again at the beginning. This display list is stored in memory locations 1586 to 1577 (the first 42 bytes of Page 6 of memory), so 0 and 6 are also POKEd into locations 560 and 561 (SDLSTL). Line 5100 restores the screen display and sets the initial X-Y coordinates.

Figure 5

Display created by Listing 1. Any 12x20 block of bytes can be displayed.



The work of scrolling is accomplished by lines 5300–6200. First, the joystick is read and the variables H and V are set (according to the direction in which the joystick is pushed). A horizontal or vertical push will set one of the variables (leaving the other at zero), while a diagonal motion will set both. (H is +1 to scroll right or –1 to scroll left; V is +1 to scroll down or –1 to scroll up.)

The new X-Y coordinates are assigned in line 5800 and analyzed by the logic in line 5900. If they fall too far towards an edge of the map, scrolling is not executed, because this would result in a display of garbage data beyond the red border.

In lines 6000–6200, the program determines how many bytes to add to or subtract from each address in the display list, and makes the variable OFS (for offset) equal to 40 times the sum of the V offset and the H offset. This portion of the program combines the techniques we discussed for horizontal and vertical scrolling above. (Remember that for vertical scrolling, you add or subtract the line length; for horizontal scrolling, you add or subtract the number of bytes along the line.)

Next, the machine-language subroutine is invoked, using SCRP$ if the offset is positive or SCRN$ if the offset is negative. These tiny programs duplicate the loop shown above, but at a much faster speed. They fly through the display list, adding or subtracting the offset from each of the twelve addresses, just as we did in BASIC. Again, Listing 3 contains the source code for each of these subroutines. Line 6200 updates the X-Y coordinates and then reads the joystick again.

CONCLUSION

We've learned that horizontal scrolling usually requires machine language's speed, but that you can achieve vertical scrolling at a tolerable rate in BASIC. Although all of the examples here use a Graphics Mode 2+16 screen, any kind of display can be scrolled. For example, suppose you want to create a picture-editing program using Graphics Mode 8. A Graphics Mode 8 screen is identical to four Graphics Mode 6 screens that are arranged two by two. Therefore, you can do detail work with a scrolling Graphics-Mode-6 display list.

If you're interested in creating maps that are similar to the one in Listings 1 and 2, you should consider Mapmaker from the Atari Program Exchange (APX). I used a modified version of its map-symbol character set for this article and created the four-screen map display with its editor.

The creation of scrolling and custom-display-list effects requires, above all, patient experimentation. Keep at it until the screen looks exactly the way you want it to!

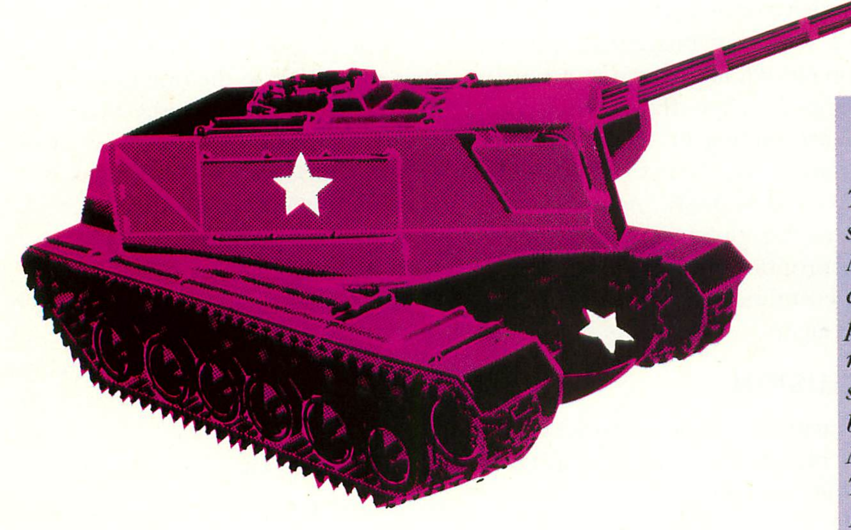
Chris Chabris graduates this month from Bryam Hills High School in Armonk, New York. He will matriculate at Harvard College this fall.

continued on page 91

USE BASIC TO AN

An easier way to program your own

by FRED PINHO



SYNOPSIS

*This article, the first in a two-part series, shows you how to control Player/Missile graphics from BASIC. The accompanying BASIC program is a four-player, tank-battle game. It requires a minimum of 32K RAM. XL owners should modify the program as specified below. For an introduction to Player/Missile graphics, see "Player/Missile Tutorial" by Chris Chabris (**Antic**, p. 14, September 1983). Part Two will appear next month.*

This tutorial was inspired by a plea from a desperate reader. Despite having read numerous magazine articles about Player/Missile (P/M) graphics, he hasn't been able to fulfill his desire: to write a tank-warfare game that uses four tanks on the screen. More importantly, he needs to learn a simple method, via BASIC, to move players and fire missiles horizontally.

Alas, it's not that simple. BASIC knows little or nothing of the P/M system, so it's not easy to control P/M graphics with BASIC. The programmer must keep track of all important registers, and must write a program to move players around on the screen. The horizontal movement of a player can be accomplished with relative ease by means of a single POKE, but vertical motion is much more complex.

You must relocate data bytes in memory to move a player vertically. And to make things even more confusing, if you move data upwards in memory, your player moves downward on the screen. As a result of these factors, vertical motion from BASIC is almost always excruciatingly slow.

You can achieve rapid player motion with machine language. However, most BASIC programmers aren't familiar enough with machine language to accomplish this.

Now for the good news. There is a way to do a passable job of achieving vertical player motion through BASIC, if you use some specialized tricks. But because of the speed problem, you'll have to keep your game fairly simple.

The accompanying program is a simple, four-player tank game. Each player is controlled by a separate joystick (one each in Ports 1 to 4). You can fire vertically and horizontally. To prevent random firing, each tank is limited to 30 shells. The first tank to hit its opponents 10 times wins. Only the score is displayed; you have to keep track of how many shots you've fired. There's a "reload" time of two to three seconds between each shot, during which you must maneuver for safety. You can only fire while moving, and you cannot move or fire diagonally.

FOR XL OWNERS ONLY

Since Atari XL machines have only two joystick ports, you cannot play Tank Battle as a four-player game. As a result, you must disable two of the tanks. The simplest way to do this is to delete lines 1170 and 1180. If you follow the program logic, this may seem to be illogical, since it eliminates the instructions for Players 1 and 2, not Players 2 and 3. (The **Antic** staff tried eliminating the instructions for Players 2 and 3, but when we did so the remaining tanks wouldn't fire.)

MODIFYING THE GAME

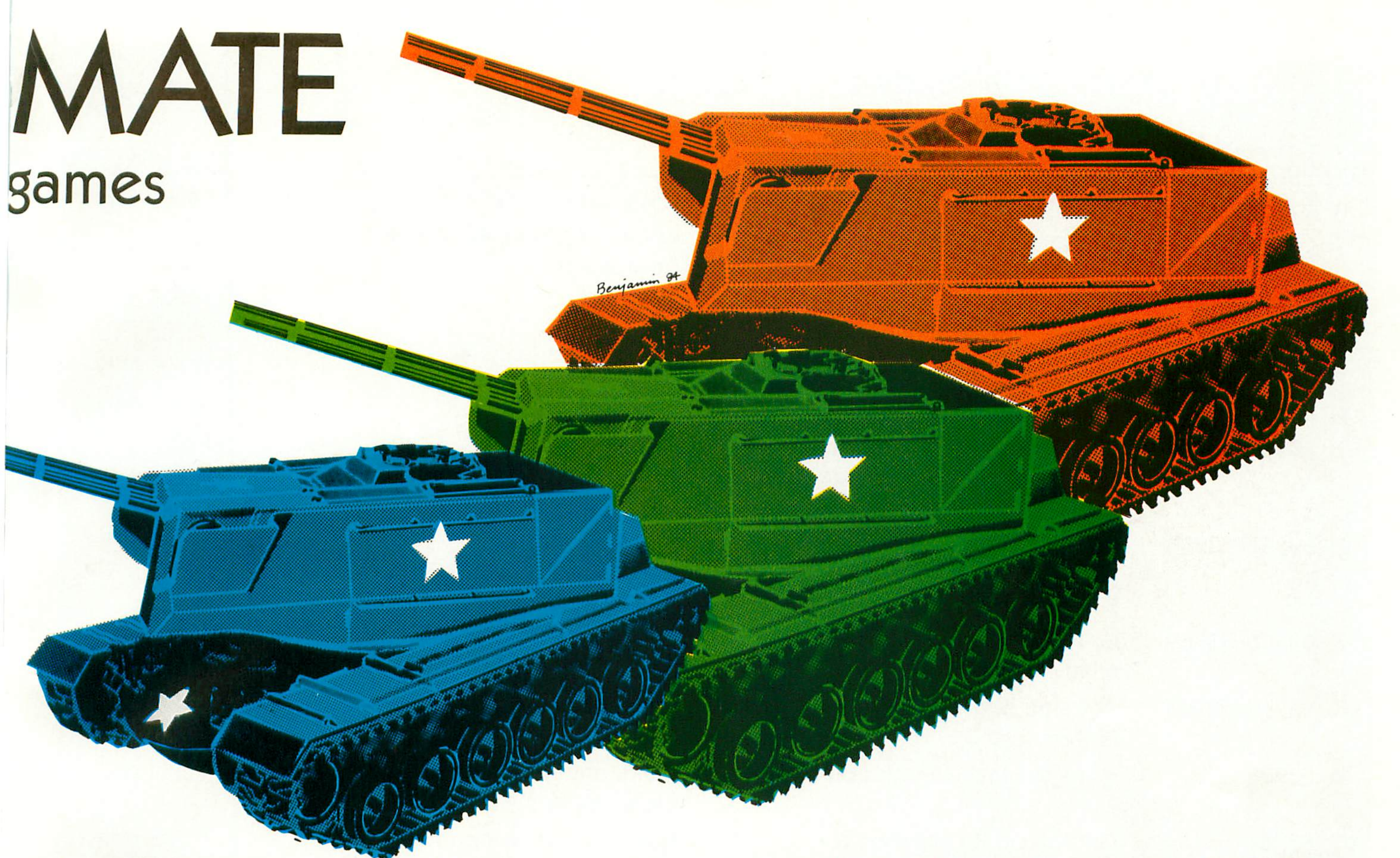
You can add features such as diagonal movement/firing and the ability to keep track of the number of shots fired without too much trouble. However, each feature you add will slow down the game. You can then move tanks and missiles in greater increments to compensate for the lack of speed, but their motion will appear "jerky," and problems with collision detection will be introduced. The choice is up to you, the programmer.

The game isn't as smooth as it would be if it were coded in machine language. However, within BASIC's limits, it's fast enough to result in a playable game.

P/M SYSTEM PARAMETERS

To help you set up housekeeping for the P/M system, I've

MATE
games



listed all of the important registers and their functions in Table 1. Note that some of the registers have two different meanings, depending on whether you're reading (with a PEEK) or writing (with a POKE) to them. For instance, you can POKE player 0's horizontal position into location 53248. But you can't check its position by PEEKing 53248. When PEEKed, that register functions as a collision register for player 0 missiles. Table 1 indicates whether a register is read only, write only, or both.

PROGRAMMING THE P/M SYSTEM

This section describes, in order, the registers that you must use to set up a Player/Missile system.

VERTICAL PLAYER RESOLUTION

See Table 1, Item 15. A player can be drawn in either of two resolutions. Single-line resolution provides high resolution, equivalent to that of Graphics 8, but uses more memory (2K). Double-line resolution, which is equivalent to that of Graphics 7, uses only 1K.

STORAGE OF P/M DATA

See Table 1, Item 14. Examine Figure 1, which shows the organization of memory for P/M data. Note that a block of memory that is safe from other activities must be reserved. My program places data below the display list, as shown in Figure 1. If you store your data here, be careful not to let the BASIC program move into the P/M data area.

In Figure 1, note that each player has its own memory space, but that the missiles are lumped together in a common area. Each player is eight-bits wide, but the missiles are only two-bits wide. Thus, each byte contains data for all four missiles. To put a missile on the screen, POKE these numbers into the missile data area:

MISSILE #	HALF-WIDTH MISSILE	FULL-WIDTH MISSILE
0	1 or 2	3
1	4 or 8	12
2	16 or 32	48
3	64 or 128	192

Data for one missile is independent of that for other missiles. You can, for instance, POKE a three into one byte for missile 0 and 12 into another byte for missile 1. Since the bytes are different, each missile appears at a different vertical location on the screen. Each missile also has its own horizontal-position register, so you have complete control over its movement.

Table 2 shows you where to store P/M data for each graphics mode, while taking certain restrictions into account. The first of these is the actual amount of memory required (2K for single-line resolution, 1K for double-line resolution). The second is that the block of memory must start on a 2K boundary for single-line resolution, and on a 1K boundary for double-line resolution. This means that the address of the start of P/M memory must be exactly divisible by 1024 (1K boundary) or 2048 (2K boundary). If you don't place P/M data on the proper boundary, the system won't work correctly.

continued on next page

TABLE 1

Player/Missile Registers

Location(s)	Function	Comments	POKE and PEEK
1) 53248–53251	Horizontal-position registers for players 0 to 3	POKE values of 0 to 255. Only values of about 48–208 are visible on the screen.	POKE only
2) 53252–53255	Horizontal-position registers for missiles 0 to 3	Same	POKE only
3) 53256–53259	Size registers for players 0 to 3	See Note 1	POKE only
4) 53260	Size register for missiles	See Note 2	POKE only
5) 53248–53251	Collision registers between missiles and playfield graphics. Missiles 0 to 3	See Note 3	PEEK only
6) 53252–53255	Collision registers between players and playfield graphics. Players 0 to 3	See Note 4	PEEK only
7) 53256–53259	Collision registers between missiles and players. Missiles 0 to 3	See Note 5	PEEK only
8) 53260–53263	Collision registers between players. Players 0 to 3	See Note 6	PEEK only
9) 53278	Used to clear all collision registers	POKE any number but zero to clear register. See Note 7	POKE only
10) 53261–53264	Graphics-data registers for players 0 to 3	Used by the computer to display P/M data. Not normally used by programmers.	POKE only
11) 53265	Graphics-data register for all missiles	Same	POKE only
12) 704–707	Color register for players 0 to 3 and associated missile	Missile has same color as its player. Must POKE color value into register. See Note 8.	Both
13) 623	Priority register	See Note 9	Both
14) 54279	PMBASE. Page number of start of Player/Missile memory area		Both
15) 559	DMA control. Specifies type of P/M resolution and whether the screen is turned on or off.	See Note 10	Both
16) 53277	Graphics control. Enables the Player/Missile system	POKE zero to disable P/M system. POKE 3 to enable P/M.	POKE only

NOTES

1. Player-Size Register

Value to POKE	Size
0,2	Normal Width
1	Double Width
3	Quadruple Width

The default (normal) register value is zero. Only the width, not the height, is affected.

2. Missile-Size Register

	Value to POKE for Size		
Missile	Normal	Double	Quadruple
0	0	1	3
1	0	4	12
2	0	16	48
3	0	64	192

Each missile's width can be controlled independently with this register.

3. Missile/Playfield Collision Registers

A playfield object is anything drawn on the screen with PLOT and DRAWTO commands. Characters that are PRINTed to the screen are also playfield objects. The playfield's number corresponds to the COLOR command used before graphics plotting. If a collision between a missile and a playfield object occurs on screen, PEEKing these registers tells you which playfield was hit.

Value in Register	Playfield Hit	COLOR Command Corresponding to Playfield
1	0	1
2	1	2
4	2	3
8	3	None. Used to print in Graphics 1 and 2

4. Player/Playfield Collision Register

These registers are affected when a player collides with a playfield. This table is identical to the one in Note 3.

5. Missile/Player Collision Registers

These registers are affected when a player collides with a missile.

Value in Register	Player Hit
1	0
2	1
4	2
8	3

6. Player/Playfield Collision Registers

These registers are affected by collisions between players. This table is identical to that in Note 5.

7. Collision-Clear Register

POKE this register with any value from zero to 255 to clear all of the collision registers. It's important that your program clear these registers frequently. If this isn't done, multiple collisions may result in unanticipated values that confuse your program. It's best to clear these registers just before each player or missile movement.

8. Player/Missile Color Registers

Set each player's color by POKEing the proper register with the result of this formula: COLOR = LUMINANCE + 16*HUE

9. Priority Register

This selects the screen objects that are to be displayed "in front" of other objects. Background color always has the lowest priority.

POKE	Priority Order
1	Players 0, 1, 2, 3, Playfields 0, 1, 2, 3
2	Players 0, 1, Playfields 0, 1, 2, 3, Players 2, 3
4	Playfields 0, 1, 2, 3, Players 0, 1, 2, 3
8	Playfields 0, 1, Players 0, 1, 2, 3, Playfields 2, 3

If you increase the value of the contents of the Priority Register by 32, you can combine players (0 with 1, and 2 with 3) to form multicolored players.

10. Direct-Memory-Access (DMA) Control Registers

This register controls the screen and Player/Missile display as follows:

Option	Add to Final Value for POKE	Comments
Narrow Playfield	1	Choose
Standard Playfield	2	only
Wide Playfield	3	one
Enable-Missile DMA	4	
Enable-Player DMA	8	
Double-Line Player Resolution	0	Default value
Single-Line Player Resolution	16	
Regular-Graphics DMA	32	

The default (normal) value for this register is 34 (regular graphics, standard playfield, P/M not enabled). POKE 559,0 to turn off the screen and speed processing time by up to 30 percent (for fast initialization).

TABLE 2

Locating P/M Data Beneath the Display List

Graphics Mode	Locate PMBASE at Indicated Offset (in pages) below RAMTOP: Double-line Resolution	Single-line Resolution
0	8	16
1	8	16
2	8	16
3	8	16
4	8	16
5	12	16
6	16	24
7	24	32
8–11	36	40

PMBASE is the location in memory of the start of P/M data. Store the page number (address/256) of PMBASE in location 54279. RAMTOP, in location 106, holds the number of pages of RAM in the machine. Let's pick Graphics 5. We code the following:

```
PM = PEEK(106)–16:POKE 54279, PM
```

So far, we've covered some of the registers listed in Table 1. I'll describe the remaining registers and provide a detailed program description next month.

Fred Pinho is a biochemical research engineer and a self-taught programmer who is interested in BASIC and assembly language. The Atari 800 is his first computer.

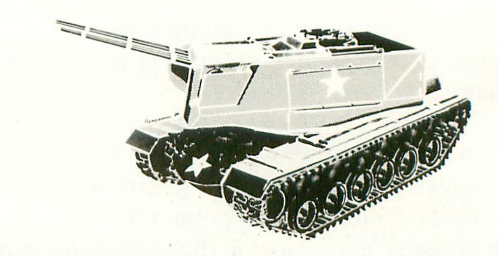
continued on next page

Figure 1
RAM Organization

Top of RAM — Top of RAM Memory
Display Data
Display List
Player/Missile Data

	Single-Line (Requires 2K)	Double-Line (Requires 1K)	
+2048			+1024
+1792	Player 3	Player 3	+ 896
+1536	Player 2	Player 2	+ 768
+1280	Player 1	Player 1	+ 640
+1024	Player 0	Player 0	+ 512
+ 768	M3 M2 M1 M0	M3 M2 M1 M0	+ 384
PMBASE	Unused	Unused	PMBASE

Free RAM
RUN/TIME STACK
STRING/ARRAY TABLE (SAT)
BASIC Program Storage
VARIABLE VALUE TABLE (VVT)
VARIABLE NAME TABLE
TOKEN OUTPUT BUFFER
Disk Operating System (If Loaded)
PAGE 6 RAM (Free RAM for Programmer's Use)
Bottom of RAM — Pages 0–5 RAM (Used by BASIC and Operating System)



```
5 REM PLAYER/MISSILE TANKS
6 REM BY FRED PINHO
7 REM ANTIC MAGAZINE
1Ø DIM TØ$(1),T1$(1),T2$(1),T3$(1),MSL
$(1),UP$(14),DW$(14),LF$(14),RT$(14),M
SLØ$(13),MSL1$(13),MSL2$(13),MSL3$(13)
2Ø GOSUB 129Ø:GOTO 115Ø
3Ø HØ=HØ+4:POKE 53248,HØ:TØ$(TØ,TØ+13)
=RT$:FOR T=1 TO 5:NEXT T:IF PEEK(5326Ø
) THEN HØ=HØ-4:POKE 53248,HØ:RETURN
4Ø IF PEEK(53252) THEN HØ=HØ-4:POKE 53
248,HØ:RETURN
5Ø IF PEEK(644) OR MØ=Ø OR DØ<1Ø THEN
RETURN
6Ø DØ=Ø:MØ=MØ-1:POKE 53252,HØ+8:MSL$(T
Ø+6,TØ+6)=CHR$(3):FOR X=HØ+8 TO HØ+95
STEP 6:POKE 53252,X
7Ø IF PEEK(53256)>1 THEN POP :SØ=SØ+1:
POKE 657,1Ø:? SØ;:MSL$(TØ+6,TØ+6)=CHR$
(Ø):RETURN
8Ø IF PEEK(53248) OR X>194 THEN POP :M
```

```
SL$(TØ+6,TØ+6)=CHR$(Ø):RETURN
9Ø NEXT X:MSL$(TØ+6,TØ+6)=CHR$(Ø):RETU
RN
1ØØ HØ=HØ-4:POKE 53248,HØ:TØ$(TØ,TØ+13
)=LF$:FOR T=1 TO 5:NEXT T:IF PEEK(5326
Ø) THEN HØ=HØ+4:POKE 53248,HØ:RETURN
11Ø IF PEEK(53252) THEN HØ=HØ+4:POKE 5
3248,HØ:RETURN
12Ø IF PEEK(644) OR MØ=Ø OR DØ<1Ø THEN
 RETURN
13Ø DØ=Ø:MØ=MØ-1:MSL$(TØ+6,TØ+6)=CHR$(
3):FOR X=HØ TO HØ-87 STEP -6:POKE 5325
2,X
14Ø IF PEEK(53256)>1 THEN POP :SØ=SØ+1
:POKE 657,1Ø:? SØ;:MSL$(TØ+6,TØ+6)=CHR
$(Ø):RETURN
15Ø IF PEEK(53248) OR X<46 THEN POP :M
SL$(TØ+6,TØ+6)=CHR$(Ø):RETURN
16Ø NEXT X:MSL$(TØ+6,TØ+6)=CHR$(Ø):RET
URN
17Ø TØ=TØ+3:TØ$(TØ,TØ+13)=DW$:FOR T=1
TO 7:NEXT T:IF PEEK(5326Ø) THEN TØ=TØ-
3:TØ$(TØ,TØ+13)=DW$:RETURN
18Ø IF PEEK(53252) THEN TØ=TØ-3:TØ$(TØ
,TØ+13)=DW$:RETURN
19Ø IF PEEK(644) OR MØ=Ø OR TØ>74 OR D
Ø<1Ø THEN RETURN
2ØØ DØ=Ø:MØ=MØ-1:POKE 53252,HØ+4:FOR X
=TØ+14 TO 88 STEP 6:MSL$(X,X+6)=MSLØ$(
1,7)
21Ø IF PEEK(53256)>1 THEN POP :SØ=SØ+1
:POKE 657,1Ø:? SØ;:MSL$(X+6,X+6)=CHR$(
Ø):RETURN
22Ø IF PEEK(53248) THEN POP :MSL$(X+6,
X+6)=CHR$(Ø):RETURN
23Ø NEXT X:MSL$(X,X)=CHR$(Ø):RETURN
24Ø TØ=TØ-3:TØ$(TØ,TØ+13)=UP$:FOR T=1
TO 7:NEXT T:IF PEEK(5326Ø) THEN TØ=TØ+
3:TØ$(TØ,TØ+13)=UP$:RETURN
25Ø IF PEEK(53252) THEN TØ=TØ+3:TØ$(TØ
,TØ+13)=UP$:RETURN
26Ø IF PEEK(644) OR MØ=Ø OR DØ<1Ø THEN
 RETURN
27Ø DØ=Ø:MØ=MØ-1:POKE 53252,HØ+4:FOR X
=TØ-2 TO 16 STEP -6:MSL$(X,X+6)=MSLØ$(
7,13)
28Ø IF PEEK(53256)>1 THEN POP :SØ=SØ+1
:POKE 657,1Ø:? SØ;:MSL$(X,X)=CHR$(Ø):R
ETURN
29Ø IF PEEK(53248) THEN POP :MSL$(X,X)
=CHR$(Ø):RETURN
3ØØ NEXT X:MSL$(X+6,X+6)=CHR$(Ø):RETUR
N
31Ø H1=H1+4:POKE 53249,H1:T1$(T1,T1+13
)=RT$:FOR T=1 TO 5:NEXT T:IF PEEK(5326
1) THEN H1=H1-4:POKE 53249,H1:RETURN
32Ø IF PEEK(53253) THEN H1=H1-4:POKE 5
3249,H1:RETURN
33Ø IF PEEK(645) OR M1=Ø OR D1<1Ø THEN
 RETURN
34Ø D1=Ø:M1=M1-1:POKE 53253,H1+8:MSL$(
T1+6,T1+6)=CHR$(12):FOR X=H1+8 TO H1+9
5 STEP 6:POKE 53253,X
35Ø IF PEEK(53257)>Ø AND PEEK(53257)<>
2 THEN POP :S1=S1+1:POKE 657,18:? S1;:
MSL$(T1+6,T1+6)=CHR$(Ø):RETURN
36Ø IF PEEK(53249) OR X>194 THEN POP :
MSL$(T1+6,T1+6)=CHR$(Ø):RETURN
37Ø NEXT X:MSL$(T1+6,T1+6)=CHR$(Ø):RET
URN
38Ø H1=H1-4:POKE 53249,H1:T1$(T1,T1+13
)=LF$:FOR T=1 TO 5:NEXT T:IF PEEK(5326
1) THEN H1=H1+4:POKE 53249,H1:RETURN
39Ø IF PEEK(53253) THEN H1=H1+4:POKE 5
3249,H1:RETURN
4ØØ IF PEEK(645) OR M1=Ø OR D1<1Ø THEN
 RETURN
41Ø D1=Ø:M1=M1-1:MSL$(T1+6,T1+6)=CHR$(
12):FOR X=H1 TO H1-87 STEP -6:POKE 532
53,X
42Ø IF PEEK(53257)>Ø AND PEEK(53257)<>
2 THEN POP :S1=S1+1:POKE 657,18:? S1;:
MSL$(T1+6,T1+6)=CHR$(Ø):RETURN
43Ø IF PEEK(53249) OR X<46 THEN POP :M
SL$(T1+6,T1+6)=CHR$(Ø):RETURN
44Ø NEXT X:MSL$(T1+6,T1+6)=CHR$(Ø):RET
URN
45Ø T1=T1+3:T1$(T1,T1+13)=DW$:FOR T=1
TO 7:NEXT T:IF PEEK(53261) THEN T1=T1-
3:T1$(T1,T1+13)=DW$:RETURN
46Ø IF PEEK(53253) THEN T1=T1-3:T1$(T1
,T1+13)=DW$:RETURN
47Ø IF PEEK(645) OR M1=Ø OR T1>74 OR D
1<1Ø THEN RETURN
48Ø D1=Ø:M1=M1-1:POKE 53253,H1+4:FOR X
=T1+14 TO 88 STEP 6:MSL$(X,X+6)=MSL1$(
1,7)
49Ø IF PEEK(53257)>Ø AND PEEK(53257)<>
2 THEN POP :S1=S1+1:POKE 657,18:? S1;:
MSL$(X+6,X+6)=CHR$(Ø):RETURN
5ØØ IF PEEK(53249) THEN POP :MSL$(X+6,
X+6)=CHR$(Ø):RETURN
51Ø NEXT X:MSL$(X,X)=CHR$(Ø):RETURN
52Ø T1=T1-3:T1$(T1,T1+13)=UP$:FOR T=1
TO 7:NEXT T:IF PEEK(53261) THEN T1=T1+
3:T1$(T1,T1+13)=UP$:RETURN
53Ø IF PEEK(53253) THEN T1=T1+3:T1$(T1
,T1+13)=UP$:RETURN
54Ø IF PEEK(645) OR M1=Ø OR D1<1Ø THEN
 RETURN
55Ø D1=Ø:M1=M1-1:POKE 53253,H1+4:FOR X
=T1-2 TO 16 STEP -6:MSL$(X,X+6)=MSL1$(
7,13)
56Ø IF PEEK(53257)>Ø AND PEEK(53257)<>
2 THEN POP :S1=S1+1:POKE 657,18:? S1;:
MSL$(X,X)=CHR$(Ø):RETURN
57Ø IF PEEK(53249) THEN POP :MSL$(X,X)
=CHR$(Ø):RETURN
58Ø NEXT X:MSL$(X+6,X+6)=CHR$(Ø):RETUR
N
59Ø H2=H2+4:POKE 5325Ø,H2:T2$(T2,T2+13
```

continued on next page

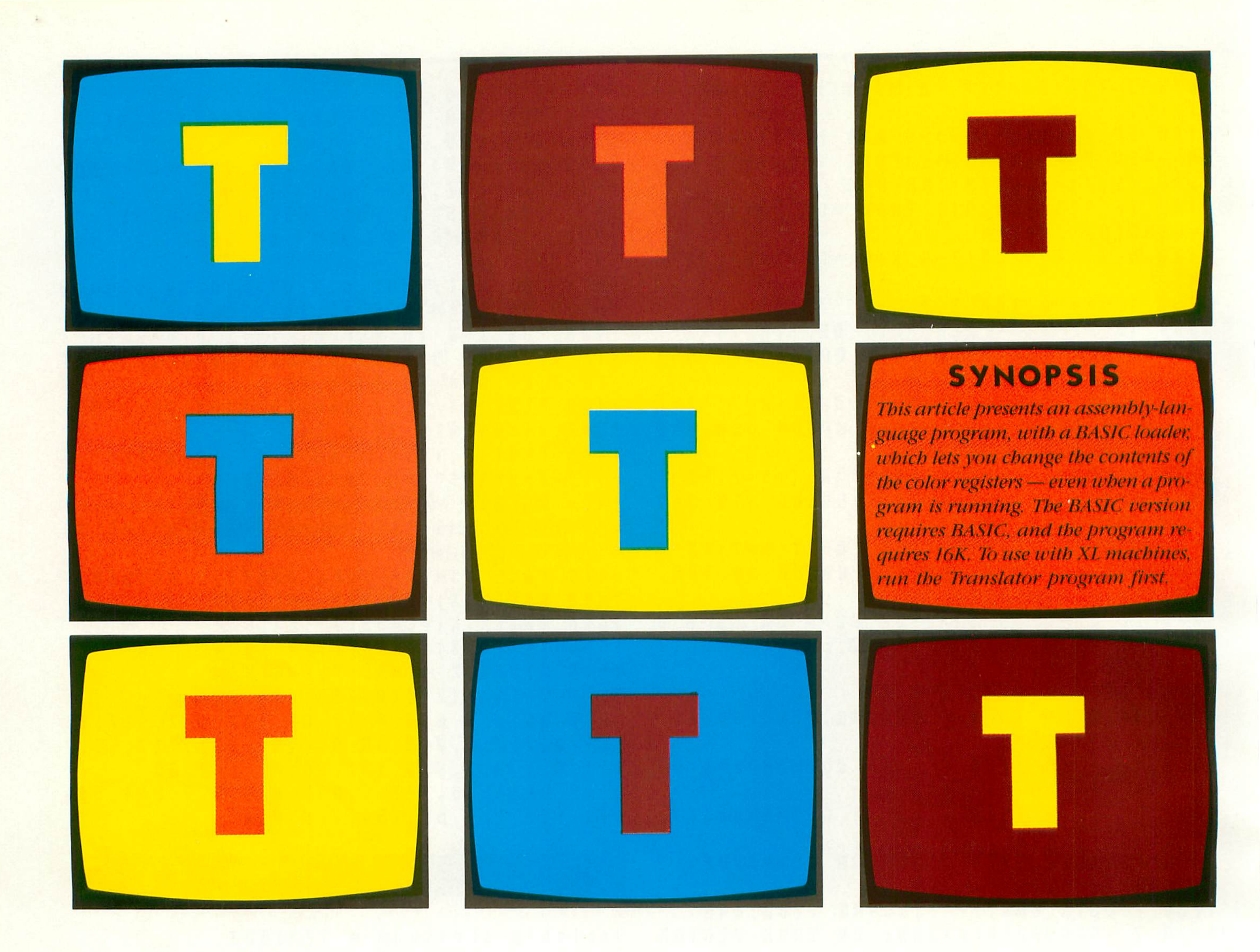
```
)=RT$:FOR T=1 TO 5:NEXT T:IF PEEK(5326
2) THEN H2=H2-4:POKE 53250,H2:RETURN
600 IF PEEK(53254) THEN H2=H2-4:POKE 5
3250,H2:RETURN
610 IF PEEK(646) OR M2=0 OR D2<10 THEN
 RETURN
620 D2=0:M2=M2-1:POKE 53254,H2+8:MSL$(
T2+6,T2+6)=CHR$(48):FOR X=H2+8 TO H2+9
5 STEP 6:POKE 53254,X
630 IF PEEK(53258)>0 AND PEEK(53258)<>
4 THEN POP :S2=S2+1:POKE 657,27:? S2;:
MSL$(T2+6,T2+6)=CHR$(0):RETURN
640 IF PEEK(53250) OR X>194 THEN POP :
MSL$(T2+6,T2+6)=CHR$(0):RETURN
650 NEXT X:MSL$(T2+6,T2+6)=CHR$(0):RET
URN
660 H2=H2-4:POKE 53250,H2:T2$(T2,T2+13
)=LF$:FOR T=1 TO 5:NEXT T:IF PEEK(5326
2) THEN H2=H2+4:POKE 53250,H2:RETURN
670 IF PEEK(53254) THEN H2=H2+4:POKE 5
3250,H2:RETURN
680 IF PEEK(646) OR M2=0 OR D2<10 THEN
 RETURN
690 D2=0:M2=M2-1:MSL$(T2+6,T2+6)=CHR$(
48):FOR X=H2 TO H2-87 STEP -6:POKE 532
54,X
700 IF PEEK(53258)>0 AND PEEK(53258)<>
4 THEN POP :S2=S2+1:POKE 657,27:? S2;:
MSL$(T2+6,T2+6)=CHR$(0):RETURN
710 IF PEEK(53250) OR X<46 THEN POP :M
SL$(T2+6,T2+6)=CHR$(0):RETURN
720 NEXT X:MSL$(T2+6,T2+6)=CHR$(0):RET
URN
730 T2=T2+3:T2$(T2,T2+13)=DW$:FOR T=1
TO 7:NEXT T:IF PEEK(53262) THEN T2=T2-
3:T2$(T2,T2+13)=DW$:RETURN
740 IF PEEK(53254) THEN T2=T2-3:T2$(T2
,T2+13)=DW$:RETURN
750 IF PEEK(646) OR M2=0 OR T2>74 OR D
2<10 THEN RETURN
760 D2=0:M2=M2-1:POKE 53254,H2+4:FOR X
=T2+14 TO 88 STEP 6:MSL$(X,X+6)=MSL2$(
1,7)
770 IF PEEK(53258)>0 AND PEEK(53258)<>
4 THEN POP :S2=S2+1:POKE 657,27:? S2;:
MSL$(X+6,X+6)=CHR$(0):RETURN
780 IF PEEK(53250) THEN POP :MSL$(X+6,
X+6)=CHR$(0):RETURN
790 NEXT X:MSL$(X,X)=CHR$(0):RETURN
800 T2=T2-3:T2$(T2,T2+13)=UP$:FOR T=1
TO 7:NEXT T:IF PEEK(53262) THEN T2=T2+
3:T2$(T2,T2+13)=UP$:RETURN
810 IF PEEK(53254) THEN T2=T2+3:T2$(T2
,T2+13)=UP$:RETURN
820 IF PEEK(646) OR M2=0 OR D2<10 THEN
 RETURN
830 D2=0:M2=M2-1:POKE 53254,H2+4:FOR X
=T2-2 TO 16 STEP -6:MSL$(X,X+6)=MSL2$(
7,13)
840 IF PEEK(53258)>0 AND PEEK(53258)<>
4 THEN POP :S2=S2+1:POKE 657,27:? S2;:
MSL$(X,X)=CHR$(0):RETURN
850 IF PEEK(53250) THEN POP :MSL$(X,X)
=CHR$(0):RETURN
860 NEXT X:MSL$(X+6,X+6)=CHR$(0):RETUR
N
870 H3=H3+4:POKE 53251,H3:T3$(T3,T3+13
)=RT$:FOR T=1 TO 5:NEXT T:IF PEEK(5326
3) THEN H3=H3-4:POKE 53251,H3:RETURN
880 IF PEEK(53255) THEN H3=H3-4:POKE 5
3251,H3:RETURN
890 IF PEEK(647) OR M3=0 OR D3<10 THEN
 RETURN
900 D3=0:M3=M3-1:POKE 53255,H3+8:MSL$(
T3+6,T3+6)=CHR$(192):FOR X=H3+8 TO H3+
95 STEP 6:POKE 53255,X
910 IF PEEK(53259)>0 AND PEEK(53259)<>
8 THEN POP :S3=S3+1:POKE 657,37:? S3;:
MSL$(T3+6,T3+6)=CHR$(0):RETURN
920 IF PEEK(53251) OR X>194 THEN POP :
MSL$(T3+6,T3+6)=CHR$(0):RETURN
930 NEXT X:MSL$(T3+6,T3+6)=CHR$(0):RET
URN
940 H3=H3-4:POKE 53251,H3:T3$(T3,T3+13
)=LF$:FOR T=1 TO 5:NEXT T:IF PEEK(5326
3) THEN H3=H3+4:POKE 53251,H3:RETURN
950 IF PEEK(53255) THEN H3=H3+4:POKE 5
3251,H3:RETURN
960 IF PEEK(647) OR M3=0 OR D3<10 THEN
 RETURN
970 D3=0:M3=M3-1:MSL$(T3+6,T3+6)=CHR$(
192):FOR X=H3 TO H3-87 STEP -6:POKE 53
255,X
980 IF PEEK(53259)>0 AND PEEK(53259)<>
8 THEN POP :S3=S3+1:POKE 657,37:? S3;:
MSL$(T3+6,T3+6)=CHR$(0):RETURN
990 IF PEEK(53251) OR X<46 THEN POP :M
SL$(T3+6,T3+6)=CHR$(0):RETURN
1000 NEXT X:MSL$(T3+6,T3+6)=CHR$(0):RE
TURN
1010 T3=T3+3:T3$(T3,T3+13)=DW$:FOR T=1
 TO 7:NEXT T:IF PEEK(53263) THEN T3=T3
-3:T3$(T3,T3+13)=DW$:RETURN
1020 IF PEEK(53255) THEN T3=T3-3:T3$(T
3,T3+13)=DW$:RETURN
1030 IF PEEK(647) OR M3=0 OR T3>74 OR
D3<10 THEN RETURN
1040 D3=0:M3=M3-1:POKE 53255,H3+4:FOR
X=T3+14 TO 88 STEP 6:MSL$(X,X+6)=MSL3$
(1,7)
1050 IF PEEK(53259)>0 AND PEEK(53259)<
>8 THEN POP :S3=S3+1:POKE 657,37:? S3;
:MSL$(X+6,X+6)=CHR$(0):RETURN
1060 IF PEEK(53251) THEN POP :MSL$(X+6
,X+6)=CHR$(0):RETURN
1070 NEXT X:MSL$(X,X)=CHR$(0):RETURN
1080 T3=T3-3:T3$(T3,T3+13)=UP$:FOR T=1
 TO 7:NEXT T:IF PEEK(53263) THEN T3=T3
+3:T3$(T3,T3+13)=UP$:RETURN
1090 IF PEEK(53255) THEN T3=T3+3:T3$(T
3,T3+13)=UP$:RETURN
1100 IF PEEK(647) OR M3=0 OR D3<10 THE
N RETURN
1110 D3=0:M3=M3-1:POKE 53255,H3+4:FOR
```

```
X=T3-2 TO 16 STEP -6:MSL$(X,X+6)=MSL3$
(7,13)
1120 IF PEEK(53259)>0 AND PEEK(53259)<
>8 THEN POP :S3=S3+1:POKE 657,37:? S3;
:MSL$(X,X)=CHR$(0):RETURN
1130 IF PEEK(53251) THEN POP :MSL$(X,X
)=CHR$(0):RETURN
1140 NEXT X:MSL$(X+6,X+6)=CHR$(0):RETU
RN
1150 SOUND 0,180,2,7
1160 D0=D0+1:A=PEEK(635):ON ((A=7)+2*(
A=11)+3*(A=13)+4*(A=14)) GOSUB 870,940
,1010,1080:POKE 53278,1
1170 D1=D1+1:A=PEEK(634):ON ((A=7)+2*(
A=11)+3*(A=13)+4*(A=14)) GOSUB 590,660
,730,800:POKE 53278,1
1180 D2=D2+1:A=PEEK(633):ON ((A=7)+2*(
A=11)+3*(A=13)+4*(A=14)) GOSUB 310,380
,450,520:POKE 53278,1
1190 D3=D3+1:A=PEEK(632):ON ((A=7)+2*(
A=11)+3*(A=13)+4*(A=14)) GOSUB 30,100,
170,240:POKE 53278,1
1200 IF S0>9 OR S1>9 OR S2>9 OR S3>9 O
R M0+M1+M2+M3=0 THEN ? "◳":POKE 656,1:
POKE 657,2:GOTO 1220
1210 POKE 77,0:GOTO 1160
1220 IF S0>9 THEN ? "BROWN COMMANDER!"
:GOSUB 1270:END
1230 IF S1>9 THEN ? "RED COMMANDER!":G
OSUB 1270:END
1240 IF S2>9 THEN ? "BLUE COMMANDER!":
GOSUB 1270:END
1250 IF S3>9 THEN ? "GREEN COMMANDER!"
:GOSUB 1270:END
1260 ? "STALEMATE!";:GOSUB 1280:END
1270 ? "CONGRATULATIONS ON YOUR VICTOR
Y!";
1280 FOR T=1 TO 2000:NEXT T:FOR X=0 TO
 4:POKE 53261+X,0:NEXT X:POKE 53277,0:
RETURN
1290 GRAPHICS 24:GRAPHICS 5:PM=PEEK(10
6)-12:POKE 54279,PM:PMBASE=256*PM:VT=P
EEK(134)+256*PEEK(135)
1300 SAT=PEEK(140)+256*PEEK(141):S0=0:
S1=0:S2=0:S3=0
1310 Z=PMBASE-SAT:RESTORE 1520:FOR X=0
 TO 4:VVT=VT+(8*X):READ Y:OFS=Z+Y;V3=I
NT(OFS/256):V2=OFS-256*V3
1320 POKE VVT+2,V2:POKE VVT+3,V3:POKE
VVT+4,128:POKE VVT+5,0:POKE VVT+6,128:
POKE VVT+7,0:NEXT X:Z=1
1330 FOR X=1 TO 14:READ Y:ON Z GOSUB 1
480,1490,1500,1510:NEXT X
1340 Z=Z+1:IF Z<5 THEN GOTO 1330
1350 POKE 53248,55:POKE 53249,55:POKE
53250,192:POKE 53251,192:H0=55:H1=55:H
2=192:H3=192
1360 FOR X=0 TO 3:POKE 53252+X,0:NEXT
X:M0=30:M1=30:M2=30:M3=30:D0=0:D1=0:D2
=0:D3=0
1370 T0$(28,37)=RT$:T1$(74,83)=RT$:T2$
(28,37)=LF$:T3$(74,83)=LF$:T0=28:T1=74
:T2=28:T3=74
1380 FOR X=0 TO 3:POKE 53256+X,0:NEXT
X:POKE 53260,0
1390 POKE 704,40:POKE 705,72:POKE 706,
136:POKE 707,200:POKE 712,0
1400 POKE 752,1:POKE 656,0:POKE 657,3
1410 ? "BROWN  0  RED  0  BLUE  0  GRE
EN  0";
1420 COLOR 1:PLOT 0,0:DRAWTO 79,0:DRAW
TO 79,39:DRAWTO 0,39:DRAWTO 0,0
1430 FOR X=0 TO 2:PLOT 14+X,6:DRAWTO 1
4+X,12:PLOT 63+X,6:DRAWTO 63+X,12:PLOT
 63+X,27:DRAWTO 63+X,33
1440 PLOT 14+X,27:DRAWTO 14+X,33:PLOT
38+X,14:DRAWTO 38+X,24
1450 PLOT 6,18+X:DRAWTO 12,18+X:PLOT 6
6,18+X:DRAWTO 72,18+X:NEXT X
1460 MSL0$="♥♥♥♥♥♥▐♥♥♥♥♥♥":MSL1$="♥♥♥♥
♥♥◢♥♥♥♥♥♥":MSL2$="♥♥♥♥♥♥ ♥♥♥♥♥♥":MSL3$
="♥♥♥♥♥♥♥♥♥♥♥♥♥"
1470 POKE 559,46:POKE 53277,3:RETURN
1480 UP$(X,X)=CHR$(Y):RETURN
1490 DW$(X,X)=CHR$(Y):RETURN
1500 LF$(X,X)=CHR$(Y):RETURN
1510 RT$(X,X)=CHR$(Y):RETURN
1520 DATA 512,640,768,896,384
1530 DATA 0,0,0,8,73,93,127,127,127,93
,65,0,0,0,0,0,0,65,93,127,127,127,93,7
3,8,0,0,0,0,0,0,127,28,62,254,62
1540 DATA 28,127,0,0,0,0,0,0,0,254,56,
124,127,124,56,254,0,0,0,0
```

TYPO TABLE

Variable checksum = 1281583

Line num range			Code	Length
5	–	60	QR	563
70	–	130	DN	517
140	–	200	LK	542
210	–	270	RD	507
280	–	340	NK	502
350	–	410	RG	536
420	–	480	YO	561
490	–	550	IU	526
560	–	620	AP	521
630	–	690	HQ	536
700	–	760	ME	561
770	–	830	RQ	526
840	–	900	QF	521
910	–	970	YB	536
980	–	1040	GB	561
1050	–	1110	DU	526
1120	–	1170	QH	547
1180	–	1220	XA	509
1230	–	1300	ZJ	500
1310	–	1360	TO	603
1370	–	1420	GT	505
1430	–	1510	RL	518
1520	–	1540	JM	189



COLOR FINETUNER

Adjust your colors to a "T"

by STEPHEN MALINOWSKI

Color Finetuner is a program that gives you direct control of the Atari's nine color registers, even when other programs are running. You can also use it to change GPRIOR, the register that selects the GTIA mode.

HOW TO INSTALL COLOR FINETUNER

The simplest way to install this utility is to enter and RUN the program listing at the end of this article. The program is installed on Page Six (starting at decimal location 1536); it causes the deferred vertical-blank (VB) vector to point to the routine.

If you need to use Page Six for your own purposes, add the lines in Listing 2. Modify line 100 to correspond to the graphics mode you'll be using in your own program. As a result, Color Finetuner will place the routine in high memory, just below the display list used by your application. It will also set HIMEM to protect the routine from BASIC. If you change graphics modes, however, the routine will no longer be protected, and it may be overwritten.

FOR ADVANCED PROGRAMMERS

Color Finetuner also can be used along with your own vertical-blank-interrupt routines. If your routines use only the immediate vector, the program can be used as is. If, however, your routines use the deferred vector or replace the OS VBI service routine entirely, the end of your routine should perform a jump to ORIGIN, as calculated in line 210. (You

shouldn't jump to INSTAL, which is used only by the BASIC USR function.) Color Finetuner concludes with a jump to XITVBV.

HOW TO USE THE ROUTINE

Once Color Finetuner is installed, type Listing 3 and RUN it. The program draws lines in Graphics 10 using all nine color registers (including the background), but you won't see some of the lines at first.

To change any color register, first select the register by pressing the corresponding number key:

KEY	REGISTER	DECIMAL LOCATION	HEX LOCATION
0	PCOLR0	704	2C0
1	PCOLR1	705	2C1
2	PCOLR2	706	2C2
3	PCOLR3	707	2C3
4	COLOR0	708	2C4
5	COLOR1	709	2C5
6	COLOR2	710	2C6 (background color in GR. 0)
7	COLOR3	711	2C7
8	COLOR4	712	2C8 (border color in GR. 0)
9	GPRIOR	623	26F

Then use the three console keys to increase or decrease the contents of the register:

OPTION — Increases the contents.
SELECT — Fast speed (hold with [START] or [OPTION]).
START — Decreases the contents.

You don't have to hold down the number key. Once you've selected a register, it will be affected by the console keys until you select another register.

DISASSEMBLY OF COLOR FINETUNER

This section contains an assembly-language listing of the Color Finetuner routine. The listing is in Atari Assembler Editor syntax and is complete, but line numbers have been deleted for editorial purposes. If you wish to type in the assembly-language program, use the command NUM to provide line numbers.

OS AND HARDWARE LOCATIONS USED BY COLOR FINETUNER

```
*=$0600         ; Sets origin to Page Six.
CONSOL=$D01F    ; Contains the composite output of the
                  console keys. The following three masks
                  are used to isolate the bit that corresponds
                  with the key:
REVMSK=$01      ; Mask for [OPTION] key.
FSTMSK=$02      ; Mask for [SELECT] key.
FWDMSK=$04      ; Mask for [START] key.
RTCLK3=$14      ; The byte of the real-time clock, which
                  is incremented every sixtieth of a second
                  (or one jiffy).
KBCODE=$D209    ; The hardware register that contains the
                  raw data for the last key pressed. It is con-
                  verted to ATASCII form by the table
                  ATASCI.
ATASCI=$FEFE    ; The table used to convert raw keyboard
                  data into ATASCII form.
RANGE=$0A       ; This number, together with TOOHI, is
                  used to convert ATASCII numbers to their
                  actual numerical values, and to check for
                  out-of-range values.
TOOHI=$C6       ; See RANGE.
PRIOR=$D01B     ; This register controls several aspects of
                  the display. See De Re Atari (Atari Program
                  Exchange, 1981) for further information.
GPRIOR=$26F     ; The OS shadow for PRIOR. Its contents
                  are written to PRIOR every sixtieth of a
                  second.
COLPM0=$D012    ; The first hardware color register.
PCOLR0=$2C0     ; The shadow register for COLPM0.
SETVBV=$E45C    ; An OS routine used to set the vectors that
                  are used as access points to the VB service
                  routine.
XITVBV=$E462    ; The last part of the OS VB service
                  routine. It restores the 6502's registers and
                  returns the processor to whatever it was
                  doing at the time the interrupt occurred.
```

THE COLOR FINETUNER ROUTINE

Every sixtieth of a second, the ANTIC chip generates an interrupt that stops the main activities of the 6502 chip and directs it to the OS vertical-blank service routine. Most of this routine is in ROM, and is not alterable by the user. However, two vectors, or "signposts," are stored in RAM. Color Finetuner (CF) causes one of these vectors to "tie into" the service routine.

The Operating System (OS) has a routine that changes these vectors. The first part of CF calls this routine:

```
INSTAL  LDA  #$7           7 equals deferred vector.
        LDX  #CHECK1/$10   high byte of the main
                           routine location.
        LDY  #CHECK1&$FF   low byte of the main
                           routine location.
        JSR  SETVBV        sets the vertical-blank
                           vectors.
        PLA                pops the stack.
        RTS                returns to BASIC.
```

Once INSTAL is called from BASIC (with a USR call), the service routine includes CF as part of its "housekeeping" routine every sixtieth of a second.

The first part of the main routine performs a series of checks to determine if CF should change a register. CHECK1 checks for a pressed console key:

```
CHECK1  LDA  CONSOL        gets console-key-register
                           data.
        CMP  #$7           is a key pressed?
        BPL  EXIT          no? then exits.
        TAY                yes? then saves CONSOL
                           value and continues.
```

continued on next page

CHECK2 controls the speed of the changes. It does this by examining the real-time clock (lowest byte at $14) and checking to see if [SELECT] has been pressed for fast color changes. At fast speed, the registers change every two cycles (one cycle = one sixtieth of a second); the normal speed changes them every sixteen cycles.

```
CHECK2  LDA  RTCLK3     gets the clock's fastest
                        byte.
        TAX             and saves it.
        AND  #$1        is it odd or even?
        BNE  EXIT       odd? then exits.
        TYA             even? then gets CONSOL
                        contents again.
        AND  #FSTMSK    is fast button pressed?
        BEQ  CHECK3     yes? then checks keyboard
                        keys.
        TXA             no? then checks clock
                        byte again.
        AND  #$F        is it a multiple of
                        (decimal) 16?
        BNE             no? then exits.
```

CHECK3 determines if a valid keyboard key has been pressed (0 through 9 only). Since KBCODE contains the raw keyboard code, not ATASCII code, we need to convert the raw code using the ATASCI table.

```
CHECK3  LDX  KBCODE     gets the last key pressed.
        LDA  ATASCI,X   looks it up in the table.
        CLC             gets ready to add.
        ADC  #TOOHI     is it high enough to be a
                        decimal digit?
        BCS  EXIT       yes? then exits.
        ADC  #RANGE     is it too low?
        BCC  EXIT       yes? then exits.
        TAX             just right? saves the
                        converted digit.
```

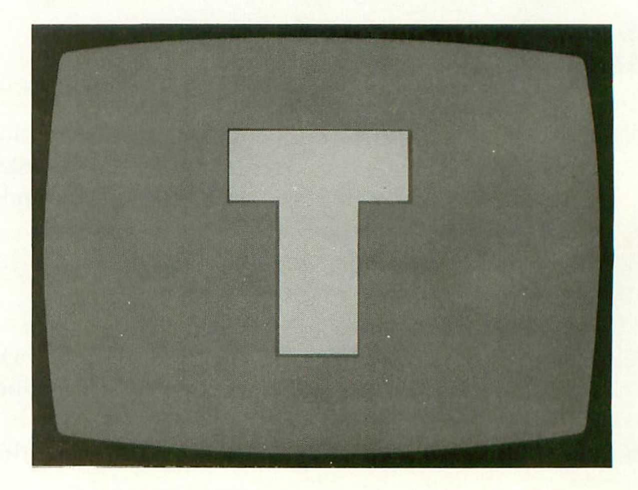
Now we're ready to change the color registers. There are actually two registers involved: the hardware register and its shadow register. If we were to write only to the hardware register, the change would last just a sixtieth of a second, because the OS copies the shadow registers to the hardware registers as part of the VB service routine.

Normally, it would be feasible to change only the shadow registers, and let the OS copy the color data to the hardware registers. However, since CF may be used in situations in which that part of the service routine is bypassed, we must write the color data to both locations.

PRIOR, the priority register, immediately follows the last hardware color register, so it can be treated as a tenth register. Its shadow GPRIOR does not immediately follow the color shadow registers, however, so it must be handled separately.

```
FORWRD  TYA             gets CONSOL contents
                        again.
        AND  #FWDMSK    is the FORWARD button
                        pressed?
        BNE  REVERSE    no? then checks reverse.
        TXA             yes? gets keyboard digit.
        CMP  #$9        is it equal to 9 (i.e.,
                        PRIOR)?
        BNE  FCOLOR     no? then it must be a col-
                        or register.
        INC  GPRIOR     yes? then increments
                        GPRIOR.
        LDA  GPRIOR     gets the incremented
                        value,
        BCS  HARD       and writes it to hardware
                        register.
FCOLOR  INC  PCOLR0,X   increments the color
                        register.
        LDA  PCOLOR0,X  gets incremented value
        BCC  HARD       and writes it to hardware
                        register.
REVRSE  TYA             gets CONSOL contents
                        again.
        AND  #REVMSK    is REVERSE button
                        pressed?
        BNE  EXIT       no? then exits.
        TXA             yes? then gets keyboard
                        digit.
        CMP  #$9        is it equal to 9 (i.e.,
                        PRIOR)?
        BNE  RCOLOR     no? then it must be a
                        color register.
        DEC  GPRIOR     yes? then decrements
                        GPRIOR.
        LDA  GPRIOR     gets decremented value,
        BCS  HARD       and writes to hardware
                        register.
RCOLOR  DEC  PCOLR0,X   decrements color register,
        LDA  PCOLR0,X   and writes it to hardware
                        register.
HARD    STA  COLPM0,X   writes new value to
                        hardware register.
EXIT    JMP  XITVBV     exits through vertical-
                        blank vector.
```

Stephen Malinowski is a free-lance musician and composer who lives in the San Francisco Bay Area. He is currently using his Atari to develop a music-animation machine that will provide visual counterparts to musical sounds.



Listing 1

```
95 REM *     COLOR FINETUNER      *
96 REM * BY STEPHEN MALINOWSKI *
97 REM *     ANTIC MAGAZINE       *
100 INSTAL=1536
200 ? "Takes a couple of seconds to in
stall.."
210 ORIGIN=INSTAL+11
220 FOR X=0 TO 107
230 READ A:POKE INSTAL+X,A
240 NEXT X
300 A=USR(INSTAL):END
1000 DATA 169,7,162,6,160,11
1010 DATA 32,92,228,104,96,173
1020 DATA 31,208,201,7,16,87
1030 DATA 168,165,20,170,41,1
1040 DATA 208,79,152,41,2,240
1050 DATA 5,138,41,15,208,69
1060 DATA 174,9,210,189,254,254
1070 DATA 24,105,198,176,58,105
1080 DATA 10,144,54,170,152,41
1090 DATA 4,208,21,138,201,9
1100 DATA 208,8,238,111,2,173
1110 DATA 111,2,176,32,254,192
1120 DATA 2,189,192,2,144,24
1130 DATA 152,41,1,208,22,138
1140 DATA 201,9,208,8,206,111
1150 DATA 2,173,111,2,176,6
1160 DATA 222,192,2,189,192,2
1170 DATA 157,18,208,76,98,228
```

Listing 2

```
100 GRAPHICS 10
110 HIMEM=PEEK(741)+256*PEEK(742)
120 INSTAL=HIMEM-110
130 HI=INT(INSTAL/256)
140 LO=INSTAL-(HI*256)
150 GRAPHICS 0
160 POKE 741,LO:POKE 742,HI
250 HI=INT(ORIGIN/256)
260 LO=ORIGIN-(HI*256)
270 POKE INSTAL+3,HI
280 POKE INSTAL+5,LO
```

Listing 3

```
10 GRAPHICS 10
20 FOR X=1 TO 32 STEP 4
30 COLOR (X-1)/4+1
40 PLOT X,0
50 DRAWTO X,190
60 PLOT X+1,0
70 DRAWTO X+1,190
80 FOR I=2 TO 3:COLOR 0:PLOT X+I,0:DRA
WTO X+I,190:NEXT I
90 NEXT X
95 GOTO 95
```

BASIC – A VARIABLE APPROACH

A variable-list utility everyone needs

by JERRY WHITE

If you program in BASIC, you're probably aware of the importance of keeping track of variables in a program. You can use the two short programs presented here to create a variable-definition chart. This chart provides a list of all of your program's variables in alphabetical order, along with a short definition of each variable. This makes your program much easier to understand.

SYNOPSIS

This article presents a variable-list utility that aids in the debugging of BASIC programs. The article's two programs require a disk drive, a printer and a minimum of 16K RAM, and run on all Atari computers.

TYPING AND STORING THE PROGRAMS

First, type in Listing 1 and leave out the REM statements. (To keep it short, I didn't include any error trapping in this program.) LIST it to disk with the following command:

LIST "D:VARIFILE.LST"

Next, type in Listing 2, and SAVE it with this command:

SAVE "D:VARISORT"

HOW TO USE THE PROGRAMS

First, LOAD your own BASIC program. My variable-lister program coexists with your program in RAM, so make sure that your program does not use lines 32000 to 32020. In addition, it should not use the following variable names: JP$, JF$, JD$, JR, and JC. Also, use no more than 123 variables.

ENTER the utility with this command:

ENTER "D:VARIFILE.LST"

To use the utility, type:

GOTO 32000

Variable names are displayed on the screen one at a time. String names are followed with '$' and array names are followed with '('. As each name appears, type in a brief description, then press [RETURN]. To make the chart easier to read, type at least one space at the beginning of your entry. If you use the [TAB] key, the definition field will automatically be left-justified, but this will leave less room for the description. The total entry, including variable name, description, and spaces, cannot exceed 39 characters.

When all variables have been defined, the utility closes the data file (D:VARIABLE.DAT) and runs the VARISORT program. VARISORT reads the data file into a string and sorts it by variable name into alphabetical order. You're then asked to type in a heading for the printed listing. Your heading should include the program name and revision number, as well as the date.

USING LONGER NAMES

If you want to use longer names and descriptions, it's easy to make the maximum entry length exceed 39 characters. For instance, if you want to use 80 characters, make the following changes in both programs. Change all incidences of 39 to 80, and all incidences of 38 to 79 (the field length minus one).

The number 4797, which appears twice in line 100 of the VARISORT program (Listing 2), was derived by multiplying the field length (39) by the maximum number of records (123). If you increase the field length to 80, you have to change 4797 to 9840 (80*123) in both places.

Jerry White is an institution in the world of Atari computing, and a long-time ***Antic*** *Contributing Editor.*

continued on page 60

CHART YOUR PROGRESS

SPECIALS OF THE MONTH

Atari 600XL	$189
Atari 800XL	$299
Atari 1050 Disk Drive	$399
Atari 1030 Modem	$115
1027 Letter Quality Printer	$299
1025 80 Column Printer	$329
Rana 1000 Disk Drive	$329
Percom S/S Disk Drive w/printer port	$519
Elephant Disks S/S	$19.50
Verbatim Twin Pak S/D	$ 5.25

HARDWARE SPECIALS

Printers

Leading Edge Gorilla	$189
NEC 8023A	$459
Okidata ML 82A	Call
Okidata ML 83A	Call
Okidata ML 84P	Call
Okidata ML 92P	Call
C. ITOH 8510 Prowriter	$399
C. ITOH 8510 SP	$665
Mannesmann Tally 160L	$629
Gemini 10X	$315
Mannesmann Tally Spirit	$345
Alphacom 81	$189

Monitors

Leading Edge Green Hi-Res 12"	$ 89
USI Amber	$169
NEC Green Screen	$169
Leading Edge Amber Hi-Res 12"	$ 89
Amdek Color 1+	$315

RECREATION

Q Bert/CRT	$ 35
Frogger/CRT	$ 35
Popeye/CRT	$ 35
Witness	$ 36
Planetfall	$ 36
Starcross	$ 29
Zork I, II, III	Ea. $ 29
Enchanter	$ 36
Deadline	$ 36
Temple of Apshai	$ 29
Pitstop	$ 27
Starleague Baseball	$ 23
Starleague Football	$ 23
Blue Max	$ 23
Joust	$ 43
Robotron 2084	$ 33
Pole Position	$ 36
Ms. Pac Man	$ 36
B-1 Nuclear Bomber (C)	$ 12
Midway Campaign (C)	$ 12
Nukewar (C)	$ 12
Telengard (C)	$ 16
Jumpman Jr./CRT	$ 27
Pitstop/CRT	$ 27
Flight Simulator II/D	$ 39
Night Mission Pinball	$ 20
Sammy Light Foot (CRT)	$ 20
Apple Cider Spider	$ 23

EDUCATION

Harcourt Brace SAT/D	$59
Chatterbee/D	$29
Match Wits/D	$22
Early Games/D	$22
Early Games Music/D	$22
Spellicopter	$29
Creature Creator	$29
Math Maze	$29
Alien Addition/D	$25
Meteor Multiplication	$25
Compu-Read/D	$22
Compu-Math/Fractions	$29
Compu-Math/Decimals	$29
Spelling Bee Games	$29
Speed Read +/D	$45
Spider Eater/Koala	$22
Bumble Bee/Learning Co.	$29
Snooper Troops 1, 2 ea.	$32
Face Maker/CRT	$25
Kids on Keys/CRT	$25
Alphabet Zoo/CRT	$25
Kid Writer/CRT	$25
Delta Drawing/CRT	$29
Success with Math Addition-Subtraction	$ 15
Success with Math Mult.-Div.	$ 15
Success with Math Fractions	$ 20
Success with Math Decimals	$ 20
Ernie's Magic Shapes CRT	$ 29
Big Bird Special Delivery	$ 29
PLATO French Vocab. Builder	$ 39
PLATO Spanish Vocab. Builder	$ 39
PLATO Basic Number facts	$ 39
PLATO Whole Numbers	$ 39
PLATO Decimals	$ 39
PLATO Fractions	$ 39
PDI Pre-school I.Q. Builder	$ 17
PDI Pre-school Library/cass	$ 49
PDI Let's Spell	$ 17
Aerobics/D	$ 35
Trains/D	$ 29
Up for Grabs/CRT	$ 29
Cosmic Life/CRT	$ 25
Delta Music/CRT	$ 29

BUSINESS

Atariwriter	$ 79
Visicalc	$179
Financial Wizard	$ 45
Letter Perfect	$ 69
Data Perfect	$ 79
Spell Perfect	$ 65
Spell Wizard	$ 45
Text/Spell Wizard Combo	$ 65
Bank Street Writer	$ 49
Bank Street Speller	$ 49
Homeword	$ 49
Syn File +	$ 65
Syn Text	$ 23
Syn Mail	$ 34
Home Accountant	$ 48
Tax Advantage	$ 45
Complete Personal Accountant	$ 65
Atspeller	$ 33
Diskette Mailing List	$ 21
Miles Payroll System	$119
Atari Accounting	$169
Family Finance	$ 36
File Manager 800+	$ 65
Bookkeeper	$105
MAC 65 W/OS A+	$ 59
BASIC XL	$ 72
BUG/65 w/OS A+	$ 25

Peripherals

Alien Voice Box	$ 99
Bit 3 80 column board	$269
Microtek 16K RAM	$ 74
Mosaic 64K (400)	$149
RAM POWER 128K	$369
Versa Writer tablet	$219
KOALA TOUCH TABLET	$ 72
CHALKBOARD POWER PAD/w cable	$ 85



Computer Outlet

1095 East Twain, Las Vegas, NV 89109 Mon.–Fri. 8 a.m. to 6 p.m., Sat. 9 a.m. to 5 p.m.



1-800-634-6766



Information and Inquiries: (702) 796-0296 Order Status: (702) 369-5523.

SHIPPING: For fast delivery, cashier checks, money orders or direct bank wires. Personal and company checks, allow 3 weeks to clear. C.O.D. charges: $3 minimum or 1% on orders over $300. Nevada residents add 5¾% sales tax. Shipping charges based on weight $3 minimum. APO and FPO orders: $10 minimum and 15% on all orders over $100. School and business purchase orders welcome. All returns must be accompanied by return authorization number. Call (702) 369-5523 before returning goods for replacement. Prices reflect cash discount only and are subject to change. Catalogs: .50¢ U.S., $1.00 foreign.

BASIC — A VARIABLE APPROACH continued from page 58

Listing 1

```
0 REM VARIFILE.LST
1 REM BY JERRY WHITE
2 REM ANTIC MAGAZINE
32000 GRAPHICS 0:POKE 82,0:POKE 710,0:
? ,"DEFINE VARIABLES:":? :DIM JF$(39),
JP$(39),JD$(39)
32002 JP$=" ":JP$(39)=" ":JP$(2)=JP$:J
F$="":CLOSE #1:OPEN #1,8,0,"D:VARIABLE
.DAT"
32004 FOR JC=PEEK(131)*256+PEEK(130) T
O PEEK(133)*256+PEEK(132)-1
32006 IF PEEK(JC)<128 THEN JR=JR+1:JF$
(JR,JR)=CHR$(PEEK(JC)):GOTO 32020
32008 IF PEEK(JC)>=128 THEN JR=JR+1:JF
$(JR,JR)=CHR$(PEEK(JC)-128)
32010 TRAP 32012:IF JF$="JR" OR JF$="J
P$" OR JF$="JC" OR JF$="JF$" OR JF$="J
D$" THEN 32018
32012 TRAP 40000:? JF$;:INPUT JD$:JR=J
R+LEN(JD$):JF$(LEN(JF$)+1)=JD$:IF JR>=
39 THEN 32016
32014 JF$(JR+1,39)=JP$(JR+1,39)
32016 ? #1;JF$
32018 JR=0:JF$="":JD$=""
32020 NEXT JC:CLOSE #1:RUN "D:VARISORT
"
```

Listing 2

```
0 REM VARISORT
1 REM BY JERRY WHITE
2 REM ANTIC MAGAZINE
100 DIM A$(4797),B$(39),C$(39):A$=" ":
A$(4797)=" ":A$(2)=A$:B$="":C$="":GOTO
 210
110 T=INT(T/3)+1:FOR L1=1 TO REC-T:FOR
 L2=L1 TO 1 STEP -T
120 IF A$(L2*39-38,L2*39)<=A$((L2+T)*3
9-38,(L2+T)*39) THEN 160
130 C$=A$(L2*39-38,L2*39):A$(L2*39-38,
L2*39)=A$((L2+T)*39-38,(L2+T)*39)
140 A$((L2+T)*39-38,(L2+T)*39)=C$
150 NEXT L2
160 POKE 709,L1*2:NEXT L1
170 IF T>1 THEN POKE 53279,0:GOTO 110
180 POKE 709,10:GOSUB 270:? :? "TYPE H
EADING FOR PRINTED LIST:":INPUT B$
190 TRAP 240:LPRINT B$:LPRINT :TRAP 40
000:GOSUB 270
200 FOR ME=1 TO REC:LPRINT A$(ME*39-38
,ME*39):NEXT ME:LPRINT :LPRINT REC;" V
ARIABLES":POKE 82,2
201 ? :? "BASIC":? "IS";:END
210 REC=0:OPEN #1,4,0,"D:VARIABLE.DAT"
:GRAPHICS 18:POSITION 7,4:? #6;"READIN
G"
220 TRAP 230:INPUT #1,B$:REC=REC+1:POS
ITION 9,6:? #6;REC:A$(REC*39-38,REC*39
)=B$:B$="":GOTO 220
230 T=REC:GRAPHICS 18:POSITION 7,4:? #
6;"sorting":GOTO 110
240 ? CHR$(253):? " CHECK PRINTER, PRE
SS START WHEN READY";
250 IF PEEK(53279)<>6 THEN 250
260 GOTO 190
270 GRAPHICS 0:POKE 710,0:POKE 82,0:RE
TURN
```



Tired of high prices, poor service and hidden charges you get from other mail order companies . . .

Try RISING SUN SOFTWARE!

ENTERTAINMENT

ARTWORX
Strip Poker (D) ... $25.00
S.P. Data Disks ... 20.00

ATARI (ROM only)
Centipede ... $32.00
Defender ... 32.00
Dig Dug ... 32.00
Donkey Kong ... 35.00
Eastern Front ... 35.00
Galaxian ... 32.00
Joust ... 35.00
Missle Command ... 27.00
Ms. Pac-Man ... 35.00
Pac-Man ... 32.00
Pengo ... 32.00
Qix ... 32.00
Robotron ... 32.00
Space Invaders ... 27.00
Star Raiders ... 32.00

BIG FIVE
Miner 2049er (R) ... $35.00

BRODERBUND
A.E. (D) ... $25.00
Arcade Machine (D) ... 42.00
Choplifter (D) ... 25.00
Choplifter (R) ... 32.00
Oper. Whirlwind (D) ... 25.00
Sea Fox (D) ... 21.00
Sea Fox (R) ... 28.00
Sky Blazer (D) ... 28.00

BUDGECO
Raster Blaster (D) ... $21.00

DATAMOST
Airstrike (D/C) ... $28.00
Bilestoad (D) ... 28.00
Mating Zone (D) ... 25.00

DATAMOST
Micropainter (D) ... $25.00
Pooyan (D/C) ... 21.00
Zaxxon (D/C) ... 28.00

DON'T ASK
Poker Sam (D/C) ... $20.00
S.A.M. (D) ... 42.00

EDU-WARE
Prisoner 2 (D) ... $28.00
Rendezvous (D) ... 28.00

ELECTRONIC ARTS
Archon (D) ... $28.00
Hard Hat Mack (D) ... 25.00
M.U.L.E. (D) ... 28.00
Murder on the Zunderneuf (D) ... 28.00
Pinball Construction Set (D) ... 28.00
Worms? (D) ... 25.00

EPYX
Jumpman (D) ... $28.00
Jumpman Jr. (D) ... 28.00

INFOCOM (disk only)
Deadline ... $35.00
Enchanter ... 35.00
Planet Fall ... 35.00
Starcross ... 28.00
Suspended ... 35.00
The Witness ... 35.00
Zork I ... 28.00
Zork II ... 28.00
Zork III ... 28.00

INTELLIGENT STATEMENTS
Pro. Blackjack (D) ... $49.00

LIGHTNING SOFTWARE
Master Type (D) ... $28.00

MUSE
Castle Wolfenstein ... $21.00

ODESTA
Chess (D) ... $49.00
Checkers (D) ... 35.00
Odin (D) ... 35.00

PARKER BROTHERS
Astrochase (R) ... $35.00
Chess (R) ... 42.00
Frogger (R) ... 34.00
Popeye (R) ... 35.00
Q-Bert (R) ... 35.00
Risk (R) ... 42.00
Super Cobra (R) ... 35.00
Tutankam (R) ... 35.00

ROKLAN
Deluxe Invaders (R) ... $28.00
Gorf (D) ... 28.00
Gorf (R) ... 32.00
Wizard of Wor (R) ... 32.00
Wizard of Wor (D) ... 28.00

SIERRA ON-LINE
Crossfire (R) ... $25.00
Crossfire (D/C) ... 21.00
Frogger (D/C) ... 25.00
Mission Asteroid (D) ... 20.00
Sammy Lightfoot (R) ... 27.00
Ultima II (D) ... 42.00
Ulysses (D) ... 28.00
Wiz & Princess (D) ... 25.00

HARDWARE DEALS

MODEMS
Apple Cat II ... $299.00
Micromodem II 265.00
Micromodem w/terminal prog. ... 295.00
212 Apple Cat ... 580.00

MONITORS
Amdek
Color I ... $299.00
Color II RGB ... 599.00
Color III RGB ... 399.00
RGB Card ... 149.00

USI
Pi1 9" Green ... $119.00
Pi2 12" Green ... 149.00
Pi3 12" Amber ... 159.00
Pi4 9" Amber ... 129.00
Color 1400 ... 299.00

PRINTERS
C. Itoh
GX-100 ... $229.00
Prowriter ... 399.00
Okidata
Microline 80 ... 349.00
Microline 82A ... 449.00
Microline 92 ... 549.00

STRATEGIC SIMULATIONS
Battle for Normandie (D/C) ... $28.00
Battle of Shiloh (D/C) ... 28.00
Combat Leader (D) ... 28.00
Cosmic Balance (D/C) 28.00
Cosmic Balance II (D/C) ... 28.00
Cytron Masters (D) ... 28.00
Galactic Gladiator (D) ... 28.00
Knights of the Desert (D/CO ... 28.00
Shattered Alliance (D) ... 28.00
Tigers in the Snow (D/C) ... 28.00

SYNAPSE
Blue Max (D/C) ... $25.00
Dimension X (D/C) ... 25.00
Ft. Apocalypse (D/C) ... 25.00
Necromancer (D/C) ... 25.00
Pharoah's Curse (D/C) 25.00
Shadow World (D/C) ... 25.00
Shamus (D/C) ... 25.00
Shamus (R) ... 32.00
Shamus II (D/C) ... 25.00
Survivor (D/C) ... 25.00
Zepplin (D/C) ... 25.00

THORN EMI
Hockey (R) ... $28.00
Jumbo Jet Pilot (R) ... 35.00
River Rescue (R) ... 28.00
Soccer (R) ... 35.00
Submarine Commander (R) ... 35.00

UTILITIES & LANGUAGES

ADVENTURE INTERNATIONAL
Diskey (D) ... $35.00

ATARI
Assembler Editor (R) . $45.00
Atari Basic (R) ... 42.00
Macro Assembler (R) .. 68.00
Microsoft Basic II (R) .. 68.00
PILOT (R) ... 60.00

DATASOFT
BASIC Compiler (D) .. $63.00
Lisp Interpreter (D) ... 70.00
Edit 6502 (R) ... 125.00

OPTIMIZED SYSTEMS
Basic A+ (D) ... $56.00
Bug 65 (D) ... 25.00
Mac/65 (D) ... 56.00

EDUCATION

ATARI
Conversational Languages
French (C) ... $42.00
German (C) ... 42.00
Italian (C) ... 42.00
Spanish (C) ... 42.00
Educator Kit ... 117.00
Inv. to Programming 1 ... 18.00
Inv. to Programming 2 ... 23.00
Inv. to Programming 3 ... 23.00
Juggles House (D) ... 23.00
Juggles House (C) ... 18.00
Juggles Rainbow (D) .. 23.00
Juggles Rainbow (C) .. 23.00
My First Alphabet (D) . 27.00
Programmer Kit ... 53.00
Tough Typing (cass) .. 18.00

EDU-WARE
Compu-Read (D) ... $21.00
Compu-Read (C) ... 15.00

SPINNAKER
Face Maker (D) ... $28.00
Hey Diddle Diddle (D) . 21.00
Kindercomp (D) ... 21.00
Most Amazing Thing (D) ... 28.00
Rhymes & Riddles (D) . 21.00
Snooper Troops 1 (D) . 32.00
Snooper Troops 2 (D) . 32.00

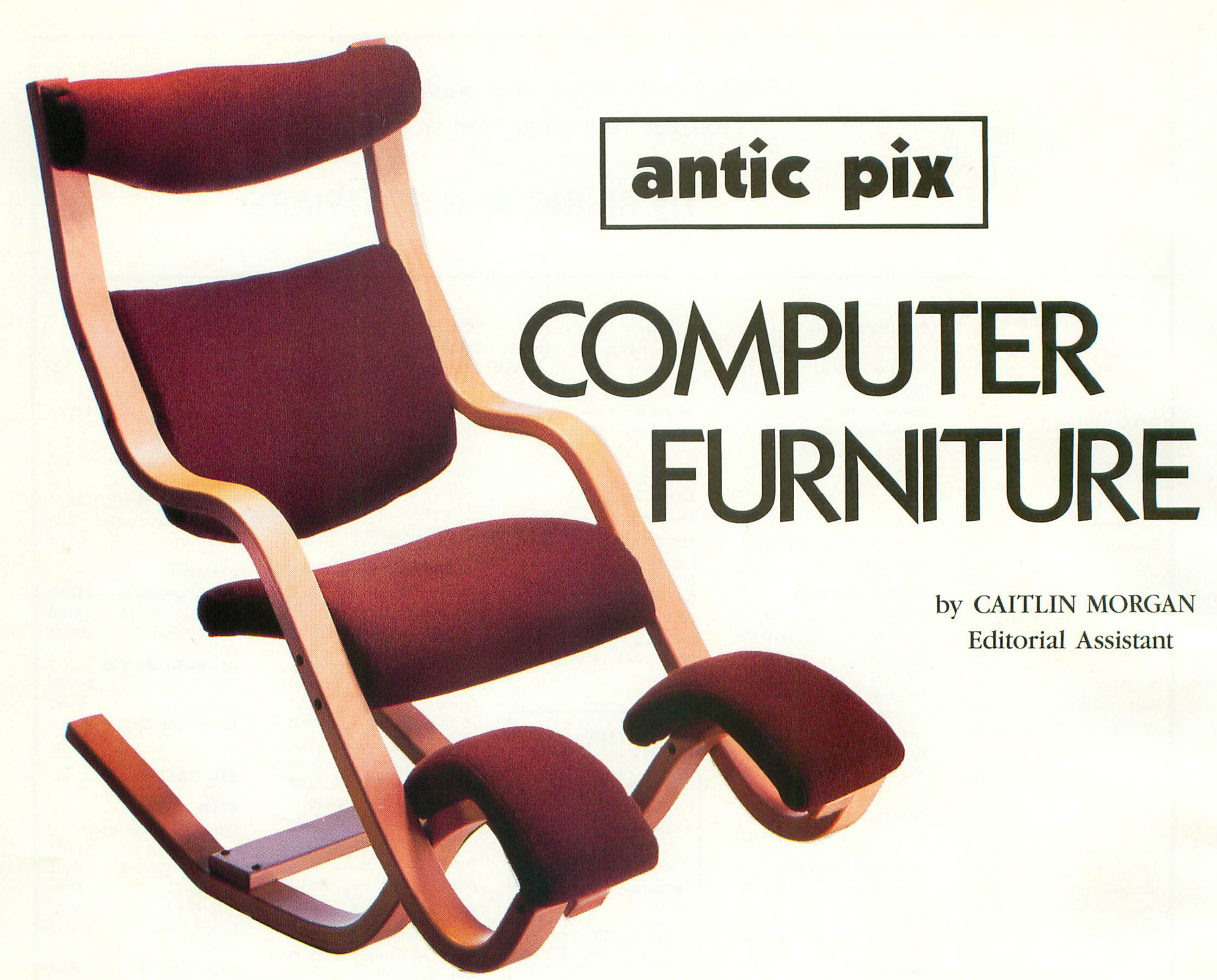
RISING SUN SOFTWARE
4200 PARK BLVD.
OAKLAND, CALIFORNIA 94602
(415) 482-3391

Ordering Information: We'll accept any form of payment—cash, personal check, money order, VISA/MasterCard, or C.O.D. Send cash at your own risk. Add $2.00 for UPS shipping; $3.00 for Blue Label Air. California residents add applicable sales tax. ALL orders shipped same day received. If we are out of stock on a particular item we will include a special bonus with your order when shipped.



CALL TOLL FREE 24 HOURS (ORDERS ONLY)
(800) 321-7770 (Outside California)
(800) 321-7771 (Inside California)





The Sharper Image's Tripos Balans Chair

antic pix

COMPUTER FURNITURE

by CAITLIN MORGAN
Editorial Assistant

Atari computer owners usually take great care in selecting software and peripherals, but their attention to the storage needs of these items often lacks a similar focus. If this dilemma sounds all too familiar, read on and discover the possibilities for organization that await you. The following furniture items and accessories have been specially designed to help tidy up your small corner of the Atari computing world.

Please note: If shipping and handling charges are not listed for a particular item, call the manufacturer to request this information.



The Drawing Board's Asymetrical Lamp

The Moore Micro Acoustical Enclosure

American Colonial Compu-Center by Bauerle Enterprises, Ltd.

The Furniture Byte's Smart Set

continued on next page

THE TRIPOS BALANS
The Sharper Image
650 Davis St.
San Francisco, CA 94111
(800) 344-4444
$499.00
Add $30.00 for shipping and handling

The **Tripos Balans** is a chair with three different personalities. In its upright position, it help's to prevent the fatigue and back strain that often accompany long hours at the computer. And after you're done programming, switch into intermediate mode. Your chair will follow, turning into an easy chair that also rocks. Finally, its full-lounge position allows you to put your feet up and luxuriate in its lush, woven-wool upholstery. Its forty pound, Norwegian-beechwood frame provides plenty of support, and assembly is easily performed with the Allen wrench included in the purchase price.

SMART SET
The Furniture Byte
P.O. Box 1757
Longview, WA 98632
(800) 426-5301
In WA — (206) 423-7277
Smart Desk DL34 — $149.00
Smart Printer Stand PL24 — $119.00
Smart Set (both of the above) — $259.00

If you believe that an intelligent machine deserves an equally sharp abode, you may be interested in **Smart Set** — a furniture duo that is at once attractive, functional and economical.

Smark Desk, at 34″ wide by 36″ tall by 24″ deep, provides plenty of storage area and leg room. And its strategically-placed shelves position both keyboard and monitor at an optimum-viewing height.

Smart Desk's counterpart, Smart Stand, can accommodate a variety of rear-feed or bottom-feed printers. Its design is said to provide substantial storage space for paper and accessories, and to help reduce irritating printer noise.

Both Smart components have doors that fold up to conceal and protect their contents, and both units are crafted of durable simulated wood. Be sure to specify which finish you prefer: Natural Oak or Walnut Woodgrain.

SERIES 580
Furniture Concepts International
720 Fifth Ave.
New York, NY 10019
(212) 586-1303
CD 580 (desk) — $199.95
CD 581 (printer stand) — $159.95
CD 582 (corner expander) — $74.95
CC 773 (chair) — $129.95

The folks at F.C.I. believe that design has been slighted in the realm of computer furniture. Their remedy is the **Series 580 Work Station**. This top-of-the-line, modular system is constructed of wood solids, wood veneers and melamine, and is said to be more durable and less scratchable than the more traditional vinyl systems on the market. All three components of the system — the desk, printer stand and corner expander — feature slanting oak borders, which add to the user's comfort. The desk and printer stand also provide cord storage areas.

For those who would rather sit than stand while computing, the CC 773 tweed chair might be just the thing. Its height and backrest are both adjustable.

MOORE MICRO ACOUSTICAL ENCLOSURE
Moore Business Center
P.O. Box 20
1400 South Wolf Rd., Suite 300
Wheeling, IL 60090
(800) 323-6230
In IL — (312) 520-3245
Starts at $139.00

You can now experience the same level of printer-noise reduction at home that is enjoyed by many businesses, thanks to the **Moore Micro Acoustical Enclosure**. Patterned after the larger enclosures commonly used in offices, this home-version surrounds your printer with a sound-absorbent foam liner and a quarter-inch-thick acrylic top. The clear, hinged cover makes all printouts visible and allows for easy access to the printer itself. The enclosure can rest directly on a work-station countertop, or it can be raised to accommodate bottom-feed printers by attaching the accompanying hardwood legs. Entry for cables, power cord and outfeed of printout is supplied by the slotted rear panel. This unit is available in four different sizes that vary in price, so be sure to call Moore for assistance in determining which model would be best for you.

ASYMETRIC LAMP
The Drawing Board
Computer Supplies
Greenwoods Industrial Park
P.O. Box 2995
Hartford, CT 06104
(800) 243-3207
$54.95

Taking its cue from the fashion industry, which is currently touting the concept of asymetry, The Drawing Board has introduced its **Asymetric Lamp** Beyond being stylish, however, it also brings relief from the eyestrain and fatigue often caused by glare. Easily attachable to any table or work station that has a flat, one-inch overhang, this work light has been designed to illuminate your keyboard and paperwork without washing out the monitor's display. The shade and reflector can be rotated 360 degrees horizontally, which enables you to direct the beam to the area of your choice. The lamp's forty-inch, movable arm allows you to adjust the light even more to your liking. The unit comes in two earth-tone colors: Putty or Chocolate Brown.

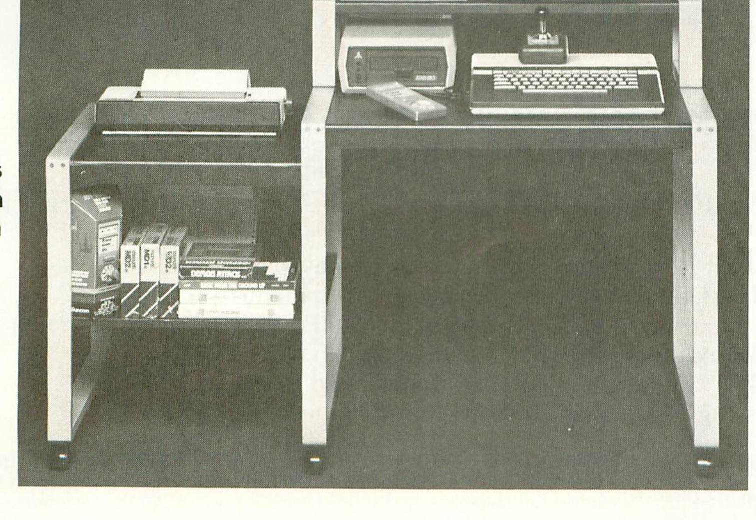
AMERICAN COLONIAL COMPU-CENTER
Bauerle Enterprises, Ltd.
P.O. Box 743
Lynbrook, NY 11563
(516) 599-1189
$169.95

If Benjamin Franklin had owned an Atari computer, he might have housed it in a piece of Americana such as this. This version of **Compu-Center** would fit in perfectly with an American Colonial decor. Hand-rubbed and oiled to emphasize its authentic-looking detail, the unit features a traditional "Trestle" design in 1 ¼″-thick pine, and can help organize all of your hardware, software and accessories. Complete instructions for assembly are included. A matching Colonial printer stand (not shown) is available for $89.95. And, for those of you with more up-to-date tastes, Bauerle Enterprises also carries more contemporary versions of Compu-Center.

Series 580 by Furniture Concepts International

Global's Tilting Monitor Stand

Stick Station



Suncom's System Station

continued on next page

SYSTEM STATION
Suncom, Inc.
650 Anthony Trail, Unit E
Northbrook, IL 60062
(312) 291-9780
Basic Unit — $79.00

The 1984 Winter Consumer Electronics Show introduced many an innovative item into the home-computer market. Among these was Suncom's **System Station**, a versatile, modular storage unit that rides on casters. Lightweight and airy in design, it is nonetheless quite capable of holding all of the elements a computer owner dreams of. The basic unit consists of a desk of comfortable, typing-table height. Add-ons include a detachable monitor stand, a printer stand and a book/diskette storage area. The desk alone is $79.00.

STICK STATION

539 W. Market St.
Louisville, KY 40202
(800) 251-3550 Ext. 3
$14.95
Add $2.50 for shipping and handling

Bring the ambiance of the arcade into your home with **Stick Station**. This hardwood joystick platform is said to help increase game-playing skills, while reducing or eliminating the pain brought on by muscle fatigue. To put the product to use, simply place a joystick into the platform's square slot and mount it with one or two screws. During game play, the station can either be set on a flat surface or held in the user's lap. Stick Station makes joystick operation a one-handed task, so your other hand is free to perform other functions, such as keyboard input. The platform is approximately 17 ½″ long by 6″ wide by 1 ½″ thick, and is configured for either the standard Atari or the Wico joystick. When ordering, be sure to specify which joystick you own. If you mention that you read about Stick Station in this article, the company will sell it to you at a bonus price of only $9.95.

THE HOME BASE — CT 125

Weber and Sons, Inc.
3468 Highway #9, RD #4
Freehold, NJ 07728
(800) 225-0044
$119.95

The Home Base — CT 125 is both a full-size computer table and a member of a family of modular computer furniture. With a work surface that is 35 ¾″ wide by 21″ deep, CT 125 can accommodate your computer and a full line of peripherals, as well as providing a storage unit for books, disks or printout paper. And when the included monitor platform is added, space is used even more efficiently. A printer-paper opening and wire harness clips are featured on the back of the desk. Adjustable floor levelers keep things in balance, and the unit's beautiful oak finish is sure to enhance the atmosphere of any home. Although The Home Base comes unassembled, it can be put together quickly with basic tools.

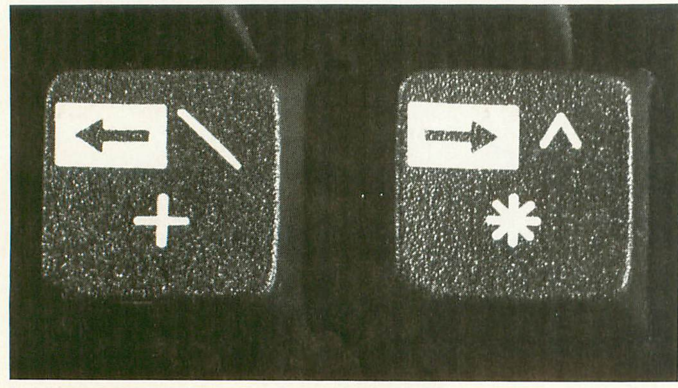
TILTING MONITOR STAND

Global Computer Supplies
Dept. AN
63 Hemlock Dr.
Hempstead, NY 11550
(800) 645-6393
In NY — (516) 292-3400
$48.00

Many maladies, such as eye fatigue and back or neck strain, are the result of long periods spent at the computer. These, and similar problems, are said to be alleviated by Global's **Tilting Monitor Stand**. Because it elevates the monitor 8″ above the desk top, and has an adjustable, twelve-inch deep shelf that tilts the monitor plus or minus ten degrees, the stand lets each user suit his or her individual need. There is also room for the computer's keyboard and accessories to be stored underneath the shelf. Finished in either oak or walnut, the stand is constructed of one-inch thick wood.

EXPLORING THE XL continued from page 40

```
5 REM OLD TO NEW OS
6 REM BY MATTHEW RATCLIFF
7 REM ANTIC MAGAZINE
10 DIM B$(FRE(0)-500),F$(20),A$(20)
20 GRAPHICS 0
30 GOSUB 1000
40 GRAPHICS 0:? :? "READING ";F$
50 TRAP 100
60 GET #2,A:B$(LEN(B$)+1)=CHR$(A):GOTO
 60
100 CLOSE #2:IF PEEK(195)=136 THEN GOT
O 200
110 ? "↖NOT END OF FILE ";PEEK(195)
120 END
200 ? "* END OF FILE *":A=0:? "Total l
ength =";LEN(B$);" BYTES"
210 ? "NOW SEARCHING FOR BAD CALLS"
```



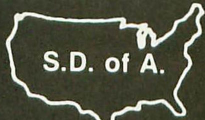
```
220 FOR I=1 TO LEN(B$)-2
230 IF B$(I,I+2)=" $v" THEN B$(I,I+2)=
" Or":? "OUTCHR FOUND AT BYTE#";I:A=A+
1
235 IF B$(I,I+2)=" >v" THEN B$(I,I+2)=
" Jr":? "GETCHR FOUND AT BYTE#";I:A=A+
1
240 NEXT I
250 ? "↖FOUND & MADE ";A;" CHANGES":IF
 A=0 THEN ? "NO CHANGES, NEEDS SOME OT
HER FIX":END
260 ? "PRESS RETURN KEY WHEN NEW DISK"
270 ? "IS READY ";:INPUT A$
280 ? "New file to write to"
290 ? "(press RETURN only if ";F$;")":
INPUT A$:IF LEN(A$)=0 THEN 310
300 F$=A$:GOSUB 1510
310 TRAP 260:OPEN #1,8,0,F$:? "Writing
 file ";F$;"....."
320 FOR I=1 TO LEN(B$):PUT #1,ASC(B$(I
,I)):NEXT I
330 CLOSE #1
340 ? "↖ALL DONE"
350 END
1000 ? "OLD ATARI OS TO  NEW 'XL OS 
XLATOR"
1010 ? "* WARNING: THIS PROGRAM MIGHT
  *"
1020 ? "* WORK, BUT IS NOT A SURE FIX
  *"
1030 ? "IT WILL READ ANY UNPROTECTED B
INARY"
1040 ? "DOS FILE AND REPLACE ALL OLD O
S"
1050 ? "CALLS TO THE PUT-CHARACTER AND
GET-"
1060 ? "CHARACTER ROUTINES TO THE NEW
ROUINES"
1070 ? "IN THE 'XL OPERATING SYSTEM."
1080 ? "  OLD OS   BECOMES  IN NEW 'X
L OS "
1090 ? "JSR $F6A4 > JSR $F2B0 OUT PUT
CHAR"
1100 ? "JSR $F63E > JSR $F24A GET KEYB
D CHAR"
1110 ? ""
1120 ? "Definitely works on the SYNASS
EMBLER."
1130 ? "I have not modified any other
progs"
1140 ? "with this,but may work. If you
 have"
1150 ? "binary program that scrolls &
locks"
1160 ? "up when should print to GR.0 s
creen"
1170 ? "or does the same when keybd in
put"
1180 ? "expected, then this fix should
 work."
1190 ? ""
1200 ? "PRESS  RETURN  to continue";:I
NPUT A$
1210 GRAPHICS 0:? :? "PUT DISK WITH FI
LE TO TRANSLATE"
1220 ? "In DRIVE #1 and press  RETURN 
";:INPUT A$
1230 OPEN #2,6,0,"D1:*.*"
1240 TRAP 1260
1250 INPUT #2,A$:? A$:GOTO 1250
1260 CLOSE #2
1270 ? "Type filename to 'xlate and"
1280 ? "press  RETURN  ";:GOSUB 1500
1290 TRAP 1300:OPEN #2,4,0,F$:RETURN
1300 ? "I/O ERROR ";PEEK(195):CLOSE #2
:GOTO 1270
1500 TRAP 1500:INPUT F$
1510 IF LEN(F$)<3 THEN ? "↖INVALID FIL
E NAME":GOTO 1500
1520 IF F$(2,2)=":" OR F$(3,3)=":" THE
N RETURN
1530 A$=F$:F$(1,3)="D1:":F$(4)=A$:RETU
RN
```

TYPO TABLE

Variable checksum = 82580

Line num range		Code	Length
5	– 120	MJ	319
200	– 290	WI	558
300	– 1050	JB	422
1060	– 1170	TI	476
1180	– 1290	GY	396
1300	– 1530	WA	226

Software Discounters of America



FOR ORDERS ONLY 1-800-225-SOFT
P.O. BOX 278-DEPT. AT • WILDWOOD, PA 15091
IN PA. (412) 361-5291

PLEASE CUT AD OUT AND USE AS SUMMER CATALOGUE

ATARI SOFTWARE

Prices Good Through August 31, 1984.

ARTWORX
Bridge 4.0 (T or D) $16
Gwendolyn (D) $18
Monkeymath (T or D) $18
Monkeynews (D) $23
Strip Poker (D) $21
Male or Female Data Disk . $16
BIG 5 SOFTWARE
Miner 2049er (R) $27
Scraper Caper (R) $33
BRODERBUND
AE (D) $21
Arcade Machine (D) $37
Bank St. Writer (D) $43
Choplifter (D) $22
Loderunner (D) $21
Mask of the Sun (D) $25
Spare Change (D) $21
CBS
Addition/Subtraction (T or D) $16
Big Bird's Special Delivery (R) $24
Decimals: Addition/ Subtraction (T or D) $16
Decimals: Multiplication/ Division (T or D) $16
Ernie's Magic Shapes (R) . $24
Fractions: Addition/ Subtraction (T or D) $16
Fractions: Multiplication/ Division (T or D) $16
Linear Equations (T or D) . . $16
Multiplication/Division (T or D) $16
Quadratic Equations (T or D) $19
Webster Word Game (R) . . $23
CONTINENTAL
Home Accountant (D) $47
Atari User's Encyclopedia (Book) $13
Book of Atari Software '84 $13

DATASOFT
Bruce Lee (T or D) $23
Dallas Quest (D) $23
Gumby (T or D) $23
Heathcliff (D) $23
Letter Wizard (D) $33
Lost Tomb (T or D) $19
Pooyan (T or D) $19
Sands of Egypt (D) $19
DESIGNWARE
Crypto Cube (D) $25
Math Maze (D) $25
Spellicopter (D) $25
Trap-A-Zoid (D) $25
EPYX
Crush, Crumble & Chomp (T or D) $19
Curse of Ra (T or D) $19
Fun w/Art (R) $25
Gateway to Apshai (R) $25
Jumpman (T or D) $25
Pitstop (R) $25
Temple of Apshai (T or D) $25

Upper Reaches of Apshai (T or D) $13
FIRST STAR
Astro Chase (T or D) $19
Boulder Dash (T or D) $19
Bristles (T or D) $19
Flip Flop (T or D) $19
GAMESTAR
Baseball (T or D) $21
Football (T or D) $21
INFOCOM
Deadline (D) $33
Enchanter (D) $33
Infidel (D) $33
Planetfall (D) $33
Sorcerer (D) $33
Starcross (D) $25
Suspended (D) $33
Witness (D) $33
Zork I, II or III (D) $25
*Free Bonus w/purchase of 3 Infocom titles
KOALA
Touch Tablet w/Software (R) $75
Touch Tablet w/Software (D) . . . . Call
Geometric Designs (D) . . . $19
Logo Design Master (D) . . . $25
Spider Eater (D) $19
LJK
Data Perfect (D) $67
Letter Perfect (D) $67
Letter Perfect (R40) $67
Letter Perfect (R80) $67
Spell Perfect (D) $55
MICROLAB
Crisis Mountain (D) $23
Death in the Caribbean (D) $25
Dino Eggs (D) $23
English Sat I, II or III (D) . . . $19
Math Sat I, II or III (D) $19
The Heist (D) $25
MUSE
Castle Wolfenstein (D) $18

Super Text Professional (D) $65
ORIGIN
Caverns of Calisto (D) $22
Exodus: Ultima III (D) $39
PDI
Alphabet Arcade (T or D) . . $16
Memory Builder (T or D) . . . $16
Number Series (T or D) $16
Preparing for the SAT (T) . . $65
Preparing for the SAT (D) . . $75
Preschool One (T or D) $16
Preschool Two (T or D) $16
Reading Comprehension (T or D) $16
Spelling Builder (T or D) . . . $16
Vocabulary One (T or D) . . . $16
Vocabulary Two (T or D) . . . $16
PARKER BROS.
Frogger (R) $33
Popeye (R) $33
Q-Bert (R) $33
*Free Bonus w/purchase of 3 Parker titles

SCARBOROUGH
Mastertype (D or R) $25
Songwriter (D) $25
SEGA
Buck Rogers (R) $25
Congo Bongo (R) $25
Star Trek (R) $25
SIERRA ON LINE
Frogger (T or D) $21
Homeword (D) $43
Oil's Well (D) $19
Quest for Tires (D or R) . . . $23
Ultima I (D) $23
Ultima II (D) $23
Ulysses (D) $21
Wizard & Princess (D) $21
SPINNAKER
Adventure Creator (R) $25
Aegean Voyage (R) $25
Aerobics (D) $27

Alphabet Zoo (R) $21
Delta Drawing (R) $25
Facemaker (R) $21
Fraction Fever (R) $21
Grandma's House (D) $21
Hey Diddle Diddle (D) $18
Kidwriter (D) $21
Kids on Keys (R) $21
Kindercomp (R) $18
Rhymes & Riddles (D) $19
Story Machine (R) $25
Trains (D) $25
SSI
Combat Leader (D) $25
Cosmic Balance (D) $25
Epidemic (D) $21
Questron (D) $33
SUBLOGIC
Flight Simulator II (D) $35
SYNAPSE
Blue Max (T or D) $21
Dimension X (T or D) $21
Encounter (T or D) $21
Ft. Apocalypse (T or D) . . . $21
Necromencer (T or D) $21
Pharoah's Curse (T or D) . . $21
Rainbow Walker (T or D) . . $21
Shamus Case II (T or D) . . . $21
Zepellin (T or D) $21
TRONIX
Chatterbee (D) $25
Pokersam (T or D) $16
S.A.M. (D) $39
ACCESSORIES
Alien Group Voice Box . . . Call
Ape Face Printer Interface $59
BASF SS, DD $17Bx
Bit 3 80 Col. Board $235
Bush Computer Furn. $25
Compuserve Starter Kit . . . $25
Disk Drive Cleaner $9
Disk Holders . . . . Call
Dust Covers for all Atari Models $6
Full Stroke Replacement Keyboard for Atari 400 . . $59
Indus. GT . . . . Call
Intec 32K Memory $49
Intec 48K Memory $89
Intec 64K Memory $109
Koala Touch Tablet . . . . . . Call
Light Pen (T or D) $25
MPP 1000 Microbits Modem $125

Printers: Alphacom, Gemini, Panasonic, Riteman, Sanyo . . . . Call
Rana 1000 . . . . Call
Sakata 13" Color Composite $239
Verbatim Disks . . . . Call
Volksmodem $59
Wico Boss $13
Wico Bat Handle $19
Wico Red Ball $21
Wico Three Way $23
Wico Trackball $29

RCA
13" diagonal Portable Color TV
Please enter me in the drawing for the 13" Color TV. (List value $329)
Name ______________________
Address ______________________
____
Drawing to be held Nov. 30, 1984. No purchase necessary. Winner will be notified by mail.

Ordering and Terms: Orders with cashier check or money order shipped immediately. Personal/company checks, allow 3 weeks clearance. No C.O.D.'s. VISA/MASTERCARD accepted with no additional charge for orders shipped to continental U.S.A. **Shipping: Continental U.S.A.** — Orders under $100 add $3; free shipping on orders over $100. PA residents add 6% sales tax. **AK, HI, FPO-APO** — add $5 on all orders. **INTERNATIONAL** — add $10 or 15% of order whichever is greatest. Defective merchandise will be replaced with same merchandise — NO CREDITS! Return must have authorization number (412) 361-5291. **Prices subject to change without notice.**

"Exodus: Ultima III, with a superior plot to match its superior gaming system, is a great game . . . it sets new standards for fantasy gaming state of the art."
Softline, November/December 1983

"Caverns of Callisto is a very challenging and enjoyable arcade game. I hope Origin Systems can continue to provide products of such quality." Core, December 1983

CAVERNS OF CALLISTO

Created by Chuckles
$34.95
Apple and Atari.

EXODUS
ULTIMA III

Created by Lord British
$59.95
Apple, Atari, Com 64 and IBM
Apple version MOCKINGBOARD enhanced.

"Exodus: Ultima III . . . is fun and exciting to play and constantly intriguing." Softalk, November 1983

ORIGIN SYSTEMS INC.

P. O. Box 99 N. Andover, MA 01845
(617) 681-0609

ESCAPE from EPSILON!

by J. D. CASTEN

Slyvester Biffdrop has been imprisoned in the bottommost of five deadly dungeons. Your task is to help him escape from all five levels. You start in Alpha, the easiest, and work your way up to the terrible Epsilon!

You can move Slyvester to the left and right with the joystick, and can make him jump with a push of the trigger. In each level, Sly begins at the leftmost point of the dungeon. To escape from each level, he must make his way to the dungeon's extreme right edge. To accomplish this, he must travel from screen to screen, overcoming obstacles, finding keys and opening doors.

The following is a list of the objects you'll encounter in the dungeons, along with a brief description of each of them:

- Spikes: If Sly touches one of these, he dies.
- Keys: To get a key, simply touch it. Keys in your possession are displayed at the bottom of the screen.



SYNOPSIS

Escape from Epsilon is a challenging arcade game for one player. The program requires a joystick, BASIC, and at least 16K RAM (24K for disk systems), and runs on all Atari computers.

- Locked Doors: If you have a key, simply touch the door. Each time you open a door, your key supply decreases by one.
- Under-Footers: Often, these are the only barrier between Sly and the dreaded spikes. You must either run along the top of these moving footways or jump over them.
- Vacuum Swings: These swing from the ceiling. To get onto one of the swings, jump up when you're directly under it. To dismount, push your joystick in the desired direction.
- Vanishing Bridges: These bridges appear and disappear unexpectedly, so you must be careful when crossing them.
- Sliding Bridges: These bridges slide back and forth over the spikes.

When Sly escapes from a given level, you're shown the number of minutes the escape took. To choose a new level, press [SELECT] and then [START].

continued on next page

```
10 REM ESCAPE FROM EPSILON
20 REM BY J.D. CASTEN
30 REM ANTIC MAGAZINE
40 GOTO 500
100 FOR Z=0 TO 1 STEP 0:IF OX<>X THEN
COLOR 32:PLOT OX,OY:PLOT OX,OY+1:OX=X:
R=1-R
105 IF OY<Y THEN COLOR 32:PLOT OX,OY:O
Y=Y
110 IF OY>Y THEN COLOR 32:PLOT OX,OY+1
:OY=Y
115 COLOR 162.5-1.5*D:PLOT X,Y:COLOR 3
.5-1.5*D+R:PLOT X,Y+1:ST=PEEK(54016):T
R=PEEK(53264)
120 IF ST=247 THEN D=1:X=X+1:LOCATE X,
Y,Q:LOCATE X,Y+1,Q1:IF Q+Q1<>64 THEN 3
00
125 IF ST=251 THEN D=-1:X=X-1:LOCATE X
,Y,Q:LOCATE X,Y+1,Q1:IF Q+Q1<>64 THEN
300
130 IF  NOT I AND TR=0 AND T<>32 THEN
I=5:Y=Y-1:R=1-R:LOCATE X,Y,Q:IF Q<>32
THEN Y=OY:I=0
135 IF  NOT I THEN 150
140 IF I THEN I=I-1:SOUND 1,5,6,I:IF
NOT I THEN 150
145 GOTO 160
150 LOCATE X,Y+2,T:IF T<>32 THEN 325
155 Y=Y+1
160 U=USR(1536):IF TM THEN GOSUB 200
165 IF GI THEN GOSUB 210
170 IF DI THEN GOSUB 220
175 LOCATE X,Y+1,Q:IF Q<>32 THEN IF Q<
160 THEN IF Q>6 THEN 375
180 NEXT Z
200 OTX=TX:COLOR 32:PLOT OTX,1:COLOR 1
1:TX=TX+DX:PLOT TX,1:IF TX=6 OR TX=13
THEN DX=-DX
202 IF X=OTX AND Y=2 THEN X=TX:I=5:D=D
X:POKE 53762,6
204 RETURN
210 IF WX<0 THEN COLOR 32:PLOT GX,5+OP
212 COLOR 9:GX=GX+WX:PLOT GX,5+OP:IF G
X=6 OR GX=14-INT(LE/1.5) THEN WX=-WX
214 RETURN
220 DT=DT+1:IF DT=20-LE*3 THEN DT=0:CO
LOR DR:PLOT 6,5:DRAWTO 13,5:DR=41-DR
222 RETURN
300 IF Q1=174 THEN 450
305 IF Q1=175 THEN IF MY THEN MY=0:GOT
O 460
310 X=OX:GOTO 150
325 IF T=10 THEN 375
330 IF OX<>X THEN POKE 53767,135:POKE
53767,0
335 GOTO 160
350 OY=Y:OX=X:FOR J=1 TO ME:COLOR 161:
PLOT J+10,22:COLOR 3:PLOT J+10,23:NEXT
 J:COLOR 32:PLOT J+9,22:PLOT J+9,23
355 IF ME=0 THEN 2000
360 GOTO 100
375 COLOR 32:PLOT OX,OY:PLOT OX,OY+1:C
OLOR 162.5-1.5*D:PLOT X,Y+1:ME=ME-1:FO
R J=30 TO 100:SOUND 1,J,10,6
380 SOUND 2,J+1,10,6:NEXT J:GOSUB 850:
X=1+17*(X>9):Y=3+6*(Y>6)+7*(Y>13):D=1-
2*(X=18):GOTO 600
400 TRAP 400:SC=SC+1-2*(X<0):X=19-OX:G
OTO 600
450 COLOR 32:PLOT KX,KY:MY=1:COLOR 174
:PLOT 9,22:KE=0
455 RESTORE 5000:FOR J=1 TO 13:READ B,
U:SOUND 1,B,10,8:FOR T=0 TO 25*U:NEXT
T:SOUND 1,0,0,0:NEXT J:NEXT Z
460 COLOR 32:PLOT 19,1:DRAWTO 19,4:PLO
T 19,8:DRAWTO 19,10:PLOT 19,15:DRAWTO
19,17:PLOT 9,22:RESTORE 4000+KH
465 READ KE,KH:RESTORE 5050:FOR J=1 TO
 17:READ B,U:SOUND 1,B,10,8:FOR T=0 TO
 25*U:NEXT T:SOUND 1,0,0,0:NEXT J
470 NEXT Z
500 A=PEEK(106)-6:GRAPHICS 17:POKE 559
,0:U=USR(ADR("hh¬Lh¬Kh¬Nh¬M"▯ ♥1K┌M◢Py
fLfNJPp◘"),57344,A*256):LE=1
505 FOR J=A*256+8 TO A*256+127:READ B:
POKE J,B:NEXT J:COM SC$(80),L$(7):Z=PE
EK(89):FOR J=1536 TO 1630:READ B
510 POKE J,B+Z*(B=1):NEXT J
550 GOSUB 850:GRAPHICS 17:GOSUB 810:DL
=PEEK(560)+256*PEEK(561):POKE DL+10,7:
POKE DL+13,7:POKE 710,0:POSITION 0,5
555 ? #6;"ESCAPE FROM EPSILON":FOR J=0
 TO 10 STEP 0.5:POKE 710,J*1.2:FOR I=0
 TO 3:SOUND I,170+I,10,J:NEXT I
560 NEXT J:U=USR(ADR("h)♥¬♣¬+i▯h_▙P_◣T
E+Pu◘")):POSITION 3,8:? #6;"by jd cast
en":FOR J=9 TO 0 STEP -0.2
565 FOR I=0 TO 3:SOUND I,170+I,10,J:NE
XT I:NEXT J:POSITION 0,12:? #6;"press
start to play"
570 RESTORE 6000+LE:READ L$,EN,KE,KH,S
C$:POSITION 3,10:? #6;"SELECT:";L$:IF
U=LE-1 THEN FOR J=0 TO 50:NEXT J
575 U=PEEK(53279):IF U=5 THEN LE=LE+1-
5*(LE=5):U=LE-1
580 IF U<>6 THEN 570
585 SC=1:X=2:Y=3:D=1:R=0:MY=0:ME=3:SOU
ND 3,5,0,0:TRAP 400:POKE 18,0:POKE 19,
0:POKE 20,0
600 SOUND 1,0,0,0:IF SC=EN THEN 700
601 GRAPHICS 17:GOSUB 810:POKE DL+26,1
34:NU=ASC(SC$(SC))
```

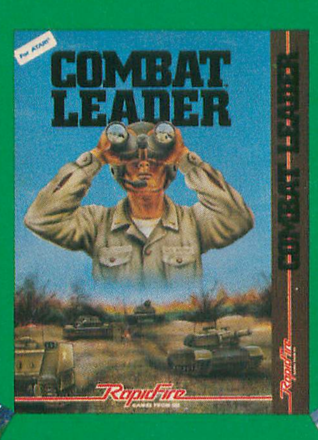
```
6Ø2 OP=Ø:IF KH=SC THEN COLOR 168:PLOT
19,1:DRAWTO 19,16:COLOR 175:PLOT 19,4:
PLOT 19,1Ø:PLOT 19,17
6Ø3 COLOR 39:PLOT Ø,Ø:DRAWTO 19,Ø:PLOT
 Ø,5:DRAWTO 19,5:PLOT 5,8:DRAWTO 14,8:
PLOT Ø,11:PLOT 1,11:PLOT 18,11
6Ø4 PLOT 19,11:PLOT Ø,14:DRAWTO 19,14:
PLOT Ø,2Ø:DRAWTO 19,2Ø:PLOT Ø,18:PLOT
1,18:PLOT 18,18:PLOT 19,18:PLOT Ø,7
6Ø5 DRAWTO 19,7:PLOT Ø,21:DRAWTO 19,21
:IF SC=1 THEN PLOT Ø,Ø:DRAWTO Ø,21
6Ø6 COLOR 136:PLOT Ø,6:DRAWTO 19,6:PLO
T 6,7:DRAWTO 13,7:COLOR 1Ø:PLOT Ø,13:D
RAWTO 19,13:PLOT Ø,2Ø:DRAWTO 19,2Ø
6Ø7 POSITION 1,22:? #6;"screens":? #6;
" to go ";EN-SC:TM=Ø:IF NU<128 THEN 61
5
611 NU=NU-128:COLOR 8:PLOT 6,Ø:DRAWTO
13,Ø:COLOR 32:PLOT 7,5:DRAWTO 12,5:PLO
T 6,6:DRAWTO 13,6:COLOR 1Ø
612 PLOT 6,7:DRAWTO 13,7:COLOR 39:PLOT
 5,6:PLOT 14,6:COLOR 11:PLOT 13,1:TM=1
:TX=13:OTX=5:DX=-1
613 IF LE>2 THEN COLOR 32:PLOT 6,5:PLO
T 13,5:IF LE=5 THEN PLOT 5,5:PLOT 14,5
615 GI=Ø:IF NU<64 THEN 62Ø
616 NU=NU-64:COLOR 32:PLOT 6,5:DRAWTO
13,5:PLOT 6,6:DRAWTO 13,6:COLOR 1Ø:PLO
T 6,7:DRAWTO 13,7:COLOR 39
617 PLOT 5,6:PLOT 5,7:PLOT 14,6:PLOT 1
4,7:GI=1:GX=6:OGX=6:WX=1
62Ø DI=Ø:IF NU<32 THEN 63Ø
621 NU=NU-32:COLOR 39:PLOT 5,6:PLOT 14
,6:COLOR 32:PLOT 6,5:DRAWTO 13,5:PLOT
6,6:DRAWTO 13,6:COLOR 1Ø:PLOT 6,7
622 DRAWTO 13,7:DI=1:DT=4:DR=9
63Ø IF NU<16 THEN 64Ø
631 NU=NU-16:IF TM=Ø THEN 635
632 COLOR 39:PLOT 6,Ø:DRAWTO 13,Ø:PLOT
 6,5:DRAWTO 13,5:PLOT 9,15:DRAWTO 9,18
:PLOT 1Ø,18:DRAWTO 1Ø,15:COLOR 32
633 PLOT 13,1:COLOR 136:PLOT 5,6:DRAWT
O 14,6:PLOT 6,7:DRAWTO 13,7:TM=Ø:GOTO
64Ø
635 COLOR 39:PLOT 5,6:DRAWTO 14,6:PLOT
 5,7:DRAWTO 14,7:COLOR 32:PLOT 12,6:PL
OT 13,6:PLOT 6,5:PLOT 7,5:PLOT 6,6
636 PLOT 7,6:PLOT 9,5:DRAWTO 9,8:PLOT
1Ø,5:DRAWTO 1Ø,8:PLOT 12,5:PLOT 13,5:C
OLOR 1Ø:PLOT 6,7:PLOT 7,7:PLOT 12,7
637 PLOT 13,7
64Ø IF NU<8 THEN 645
641 NU=NU-8:COLOR 32:FOR J=5 TO 8:PLOT
 J,J:DRAWTO 19-J,J:NEXT J:COLOR 39:PLO
T 5,6:PLOT 6,7:PLOT 14,6:PLOT 13,7
642 OP=1
645 Q=Ø:IF NU<4 THEN 65Ø
646 NU=NU-4:COLOR 39:PLOT 6,9:DRAWTO 8
,11:DRAWTO 1Ø,11:DRAWTO 8,9:COLOR 136:
PLOT 5,7:DRAWTO 8,1Ø:Q=1
65Ø IF NU<2 THEN 655
651 NU=NU-2:COLOR 39:PLOT 11,9:DRAWTO
9,11:DRAWTO 11,11:DRAWTO 13,9:COLOR 13
6:PLOT 14,7:DRAWTO 11,1Ø:Q=Q+1
652 IF Q=2 THEN PLOT 8,9:DRAWTO 11,9:P
LOT 9,1Ø:PLOT 1Ø,1Ø
655 EL=Ø:IF NU<1 THEN 66Ø
656 NU=NU-2:COLOR 32:PLOT 3,12:DRAWTO
16,12:PLOT 4,13:DRAWTO 15,13:PLOT 5,14
:DRAWTO 14,14:COLOR 136:PLOT 3,14
657 EL=1:DRAWTO 1,12:PLOT Ø,13:PLOT 1,
13:PLOT 16,14:DRAWTO 18,12:PLOT 19,13:
PLOT 18,13:FOR J=5 TO 9:PLOT J,25-J
658 DRAWTO 19-J,25-J:NEXT J:COLOR 39:P
LOT Ø,11:PLOT 1,11:DRAWTO 5,15:DRAWTO
5,15:DRAWTO Ø,14:PLOT 19,11
659 PLOT 18,11:DRAWTO 14,15:DRAWTO 19,
14:PLOT 5,2Ø:PLOT 5,19:DRAWTO 9,15:PLO
T 1Ø,15:DRAWTO 14,19:PLOT 14,2Ø
66Ø IF GI THEN COLOR 9:PLOT 6,5
661 IF EL THEN 664
662 COLOR 172:PLOT 19,12:DRAWTO 12+LE,
12:PLOT 9,12:DRAWTO 2+LE,12:PLOT Ø,19:
DRAWTO 7-LE,19:PLOT 1Ø,19
663 PLOT Ø,19:DRAWTO 7-LE,19:PLOT 1Ø,1
9:DRAWTO 17-LE,19
664 IF Q+EL=3 THEN COLOR 136:PLOT 9,11
:PLOT 1Ø,11:COLOR 39:PLOT 9,12:DRAWTO
9,14:PLOT 1Ø,14:DRAWTO 1Ø,12
69Ø IF KE=SC THEN RESTORE 3ØØØ+KE:READ
 KX,KY:COLOR 174:PLOT KX,KY
691 IF MY THEN COLOR 174:PLOT 9,22
692 IF TM+GI<2 THEN 694
693 COLOR 39:PLOT 6,Ø:DRAWTO 13,Ø:PLOT
 9,1:DRAWTO 9,6:PLOT 1Ø,6:DRAWTO 1Ø,1:
COLOR 32:PLOT 13,1:PLOT 6,5:GI=Ø:TM=Ø
694 POKE 77,Ø:GOTO 35Ø
7ØØ GRAPHICS 17:GOSUB 81Ø:POKE 711,2ØØ
:POKE 712,13Ø:COLOR 39:PLOT Ø,5:DRAWTO
 2,5:PLOT Ø,11:DRAWTO 2,11:PLOT Ø,18
7Ø5 DRAWTO 2,18:COLOR 136:PLOT 6,5:DRA
WTO 5,5:DRAWTO 5,18:PLOT 4,11:PLOT 6,1
1:PLOT 8,18:DRAWTO 8,11:DRAWTO 1Ø,11
71Ø FOR J=Ø TO 4 STEP 4:PLOT J+14,18:D
RAWTO J+12,18:DRAWTO J+12,11:DRAWTO J+
14,11:DRAWTO J+14,15:PLOT J+13,15
715 NEXT J:POSITION 8,5:RESTORE 6ØØØ+L
E:READ L$:? #6;L$:COLOR 161:PLOT Ø,Y:C
OLOR 3:PLOT Ø,Y+1:POSITION 8,7
72Ø ? #6;"TIME:";INT((PEEK(18)*65536+P
EEK(19)*256+PEEK(2Ø))/3595.3646+Ø.5):P
```

continued on page 76

All roads to the best strategy games for the ATARI® lead to SSI.

As the hero in this fantasy adventure role-playing game, you must battle hordes of deadly monsters as you seek out to destroy the evil wizard, Mantor. Use your strength, dexterity, intelligence and charisma to the fullest in your treacherous journey to save the Questron Empire. On 40K disk. $49.95.

Arcade-like graphics and action plus strategy-gaming sophistication and realism make COMBAT LEADER™ one of the greatest tank-battle games ever! You can choose from over 70 historical and modern tanks. On 48K disk and cassette. $39.95.

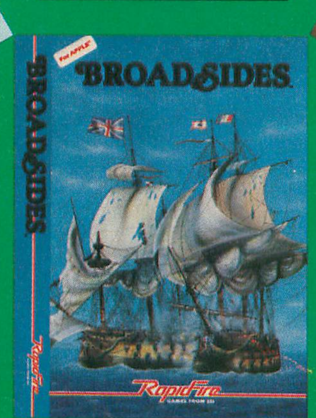


As the pilot of a World War II B-17 bomber in this role-playing game, can you survive 50 dangerous but exciting raids over France and Germany to earn the crushed cap of a true veteran? Find out as you travel back to 1942 as part of the 8th Air Force Bomber Group. On 40K disk. $39.95.



STRATEGIC SIMULATIONS INC.

15 months of development time have gone into this game to make it the definitive division-level simulation of the Russian War. You can re-enact the entire campaign from June 1941 through December 1944 or select one of 3 shorter scenarios. If you call yourself a real wargamer, you must play it! On 48K disk. $79.95.



This strategy/arcade game brings back all the romance and excitement of the adventurous age of fighting sail by letting you command your very own 18-century warship. You'll make all the military decisions a real captain must make, and you can even construct a ship of your own design. On 48K disk. $39.95.

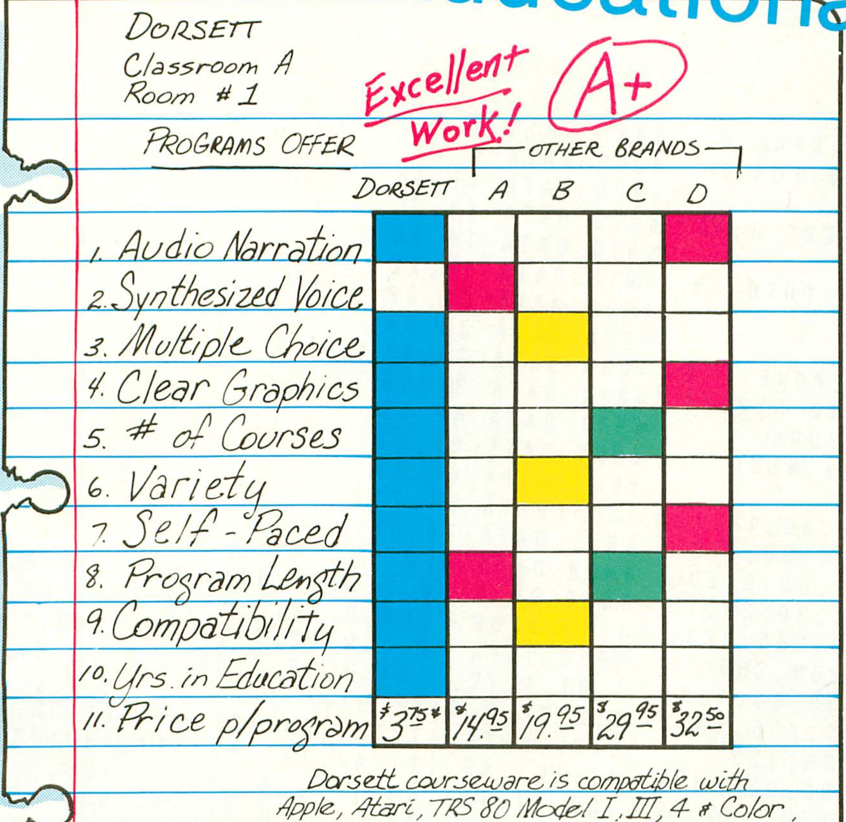
RAILS WEST!™ is a sophisticated business/educational game that takes you back to 1870. You'll wheel and deal as you try to build the richest Transcontinental railroad. For up to 8 players, it is fun, challenging, and exhaustive in detail and realism. On 40K disk. $39.95.

For all our ATARI® games, write for your free copy of SSI's color catalog.

If there are no convenient stores near you, VISA & M/C holders can order direct by calling **800-227-1617, ext. 335 (toll free).** In California, call 800-772-3545, ext. 335. To order by mail, send your check to: SSI, 883 Stierlin Road, Bldg. A-200, Mountain View, CA 94043. Please include $2.00 for shipping & handling. (California residents, add 6.5% sales tax.) **All SSI games carry a 14-day "satisfaction or your money back" guarantee.**

For Your Atari Computer
TRS 80 Color, I, III & 4 or Apple Computer

Dorsett Educational Software Passes the Cost-Efficiency Test!



DORSETT
Classroom A
Room #1

Excellent Work! A+

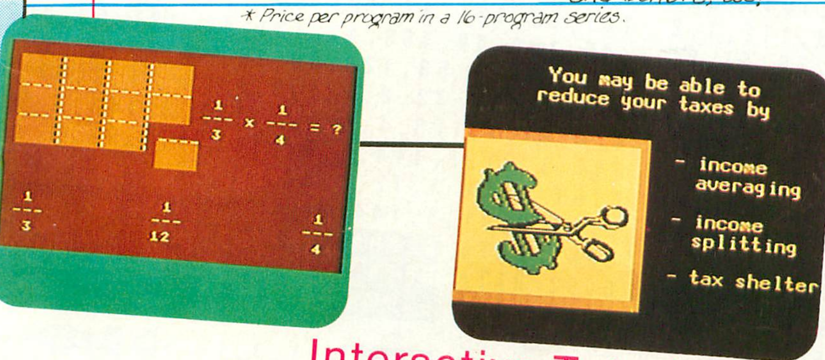
PROGRAMS OFFER

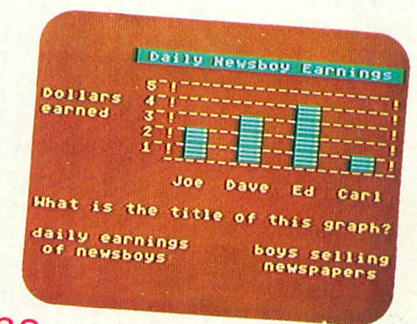
	DORSETT	A	B	C	D
		OTHER BRANDS			
1. Audio Narration	■				■
2. Synthesized Voice		■			
3. Multiple Choice	■		■		
4. Clear Graphics	■				■
5. # of Courses	■			■	
6. Variety	■		■		
7. Self-Paced	■				■
8. Program Length	■	■		■	
9. Compatibility	■		■		
10. Yrs. in Education	■				
11. Price p/program	$3.75*	$14.95	$19.95	$29.95	$32.50

Dorsett courseware is compatible with Apple, Atari, TRS 80 Model I, III, 4 & Color, and others, too!

* Price per program in a 16-program series.

For computer assisted instruction, our software more than passes the test...it EXCELS! With over 1000 educational programs designed specifically to concentrate learning for all users, from kindergarten level to advanced individual studies, from slow learners to exceptionally bright students, you'll get positive results with our quality, economically-priced courseware.





Interactive Tutorial Programs

Over 1000 Programs with Full Time Audio Narration, Pictures & Text!

We're Your Educational Software Source

Course	No. of Programs
Reading	256
Mathematics	128
Comprehension	48
History	32
Algebra	16
Spelling	16
Government	16

16 Programs in each of the following:
Carpentry - Electronics - Health Services
Office Skills - Statistics - First Aid/Safety
Economics - Business - Accounting
Psychology
AND MANY MORE!

Dealer inquiries welcome

For your Atari 400/600/800/1200, you will need the Atari Cassette Recorder and the Dorsett 4001 Educational Master Cartridge, **$9.95.** For your Apple II, you will need the Dorsett M402 T/T plug-in board, **$99.00,** and the M401 stereo cassette player, **$79.00.** All programs listed are available for TRS 80, I, III, 4, which require the M203 speaker converter, **$99.00,** and 401 stereo cassette player, **$79.00.**

$59.90 for an album containing a 16-program course (8 cassettes with 2 programs each at **$3.75** per program). **$8.80** for a 2-program cassette.

Send for a catalog of over 1000 programs for Atari, TRS 80, Apple, etc.

For more information, or to order call:

TOLL FREE 1-800-654-3871
IN OKLAHOMA CALL (405) 288-2301



DORSETT
Educational Systems, Inc.
Box 1226, Norman, OK 73070



ESCAPE FROM EPSILON continued from page 73

```
OSITION 8,9:? #6;"PRESS START"
725 RESTORE 5100:FOR J=0 TO 31:READ B,
U:SOUND 0,B,10,6:SOUND 1,U,10,6:FOR T=
0 TO 6:GOSUB 800:NEXT T:NEXT J
730 FOR J=0 TO 500:GOSUB 800:NEXT J:GO
TO 725
800 IF PEEK(53279)=6 THEN POP :GOTO 55
0
805 RETURN
810 POKE 708,246:POKE 709,122:POKE 710
,12:POKE 711,50:POKE 756,A:POKE 512,79
:POKE 513,6:POKE 54286,192:RETURN
850 FOR J=0 TO 2:SOUND J,0,0,0:NEXT J:
RETURN
900 DATA 0,0,48,104,126,60,16,56,40,10
0,56,120,112,56,112,120,104,54,56,120,
108,230,140,206,0,0,12,22,126,60,8,28
905 DATA 20,38,28,30,14,28,14,30,22,10
8,28,30,54,103,49,115,255,189,255,223,
251,255,191,236,255,255,255,255,255
910 DATA 255,255,255,255,16,40,68,130,
1,255,0,34,34,34,34,34,34,119,255,56,1
6,16,16,16,124,254,130,255,129,129
915 DATA 255,0,0,0,0,124,92,92,92,68,1
24,16,254,0,0,0,224,160,190,170,234,25
5,255,231,195,195,231,231,255,104,160
920 DATA 0,190,112,1,169,0,153,112,1,2
00,192,20,208,7,138,153,111,1,76,41,6,
185,112,1,201,140,208,8,153,111,1,169
925 DATA 0,153,112,1,76,11,6,160,20,19
0,251,1,169,0,153,251,1,136,192,0,208,
5,138,153,252,1,96,185,251,1,201,140
930 DATA 208,8,153,252,1,169,0,153,251
,1,76,51,6,72,169,28,141,10,212,141,22
,208,169,78,141,25,208,104,64
2000 POSITION 1,22:? #6;"        game over
":? #6;"      PRESS START":RESTORE 5150:
U=1:Z=1:FOR Q=1 TO 27:GOSUB 800:U=U-1
2005 IF U=0 THEN SOUND 0,0,0,0:READ B,
U:SOUND 0,B,10,6
2010 Z=Z-1:IF Z=0 THEN SOUND 1,0,0,0:R
EAD B,Z:SOUND 1,B,10,6
2015 FOR T=0 TO 25:NEXT T:NEXT Q:GOSUB
 850:FOR Q=0 TO 300:GOSUB 800:NEXT Q:G
OTO 2000
3002 DATA 19,17
3003 DATA 18,10
3004 DATA 1,10
3005 DATA 18,10
3006 DATA 1,10
3009 DATA 18,17
3011 DATA 1,10
3016 DATA 18,17
3039 DATA 1,10
3042 DATA 18,10
3056 DATA 1,10
3060 DATA 15,4
3076 DATA 18,10
4007 DATA 0,0
4008 DATA 9,15
4009 DATA 11,14
4012 DATA 13,25
4014 DATA 19,20
4015 DATA 16,22
4022 DATA 29,30
4025 DATA 28,34
4030 DATA 39,41
4034 DATA 38,40
4041 DATA 42,50
4050 DATA 56,57
4057 DATA 60,63
4063 DATA 64,71
4071 DATA 76,80
4080 DATA 0,0
5000 DATA 60,2,72,2,60,4,60,2,72,2,60,
4,53,2,60,2,64,2,72,2,81,2,72,2,64,4
5050 DATA 60,2,91,2,91,1,91,1,91,2,91,
1,81,1,72,1,64,1,60,4,60,2,81,2,81,2,6
4,2,72,2,81,2,91,4
5100 DATA 47,121,53,121,60,121,53,121,
47,121,45,121,40,121,40,121,45,108,40,
108,47,121,45,121,53,128,45,128,60
5103 DATA 144,53,128,47,121,53,121,60,
121,53,121,47,121,45,121,40,121,40,121
,45,108,40,108,47,121,45,121,53,162
5107 DATA 53,128,60,121,0,0
5150 DATA 72,2,121,8,91,4,71,2,91,3,14
4,4,91,1,108,1,128,1,121,3,144,3,91,1,
144,1,91,1,144,1,91,1,144,1,91,1,144
5163 DATA 1,72,1,121,1,72,1,121,1,81,2
,128,2,91,3,144,3
6001 DATA ALPHA   ,6,2,5,◥♥ ♥◣
6002 DATA BETA ,11,5,7,▪♥ @◣@◥▪♥P
6003 DATA GAMMA,21,4,9,I▪@▪ A♥◣◥@▪@◥♣@
▪  ▪◥
6004 DATA DELTA,41,3,12,◥♥◣♥♣▪L@A@♣@▪
!♣ !  0  @▪♥@&♥A ♥◣@L▪@▪'H
6005 DATA EPSILON,81,6,8,'P@▪◥F▪P▪H◣▪F
▪A♣@◣▪♣A2@▪◥0@▪▪'I▪▪'H@▪'▪'H◣@◣LJLJ▪'H
▪'H▪▪◥P◢A◣G◢F◣▪♣!▪▪♥&I▪A◣◣▪F▪
```

TYPO TABLE

Variable checksum = 729234

Line num range			Code	Length
10	–	125	HF	505
130	–	200	FV	479
202	–	330	WQ	451
335	–	380	UO	500
400	–	465	SQ	565
470	–	555	PV	558
560	–	585	KX	612
600	–	604	VR	595
605	–	611	EG	526
612	–	617	XH	584
620	–	633	BU	617
635	–	641	PP	575
642	–	655	OF	511
656	–	659	BK	650
660	–	692	CE	522
693	–	705	DY	556
710	–	725	JI	543
730	–	905	KH	510
910	–	930	TK	538
2000	–	3016	MA	505
3039	–	4022	MF	126
4025	–	5100	VG	363
5103	–	6005	PL	543

IMPORTANT NOTICE:

In case you didn't notice, Antic did **NOT** publish a **MAY** issue!

We didn't skip an issue, we just moved up our cover date to facilitate distribution.

This change will not affect your subscription, or the number of issues published in 1984. There will be 12 issues in this volume. Subscriptions will be adjusted automatically, and you can expect to receive each issue at least two weeks before its cover date.

new products

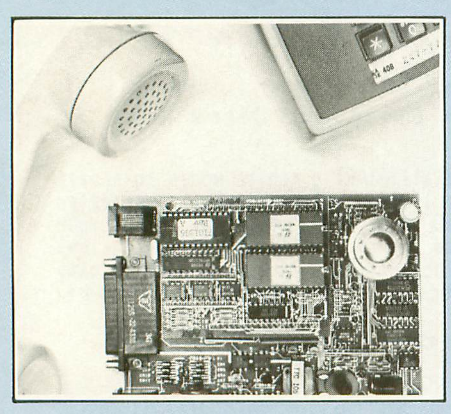
AN ATARI FOR KIDS

(book)
dilithium Press
8285 SW Nimbus, Suite 151
Beaverton, OR 97005
(800) 547-1842
In OR — (503) 646-2713
$7.95
AN ATARI IN THE CLASSROOM:
Activity Workbook — $5.95
AN ATARI IN THE CLASSROOM:
Teacher's Guide — $14.95

In **An Atari For Kids**, a circus theme is the delightful means by which a knowledge of BASIC is imparted to children ages 8–13. Three teachers collaborated on this text, which teaches good programming skills with a focus on problem solving, improved thinking skills and creativity. A complementary activity workbook and teacher's guide are available separately.

POPCOM — MODEL X100

(modem)
Prentice Corp.
266 Caspian Dr.
P.O. Box 3544
Sunnyvale, CA 94088
(408) 734-9810
$475.00



Capable of true-voice and data switching at your work station, the **Popcom X100** modem eliminates the need for separate phone lines for your telephone and computer. The task of dialing multiple calls to switch between voice and data is also made obsolete. Popcom provides complete call-progress monitoring, and can detect dial tones, busy signals, remote ringing and even line-current disconnect. It can be installed almost anywhere.

DROL

(game)
Broderbund Software, Inc.
17 Paul Dr.
San Rafael, CA 94903
(415) 479-1170
48K — diskette
$34.95



Two lost children have fallen under the spell of a witch doctor and are wandering, zombie-like, through the underground corridors of an ancient civilization. Taking the technology of the future to this antiquated world, you attempt their rescue with the aid of a rocket backpack, a laser gun and a radar scope. But beware! An amusing and unlikely collection of villains will attempt to prevent you from reuniting the children with their mom.

MICROINDEX

(reference tool)
Serious Personal Computing
P.O. Box 7059
South Nashua, NH 03060
(603) 888-1376
monthly — $99.00 per year
abridged — $49.00 per year
journal-specific annuals — $5.00–$12.00 per issue

Antic is, of course, among the magazines listed in **Microindex**, a comprehensive index to microcomputing-oriented periodicals. Article data includes title, author, page, length, journal, issue, reader level and rating. Article types include features, articles, product reviews and announcements. Microindex is intended to organize the vast amount of information disseminated in a large number of periodicals, and is designed for people of all ages and computing levels.

ALPHACOM 81

(printer)
Alphacom, Inc.
2323 So. Bascom Ave.
Campbell, CA 95008
(408) 559-8000
$16.95
Atari cable — $44.95

Alphacom's latest printer, the **Alphacom 81**, is a high-performance, dot-matrix printer that boasts an eighty-column line width and full graphics capability. It prints one hundred characters per second, and advanced thermal technology allows it to do so quietly.

THE WIRE TREE

(power source)
Networx
203 Harrison Place
Brooklyn, NY 11237
(212) 821-7555
$69.95



Engineered specifically for personal computers, **The Wire Tree** provides reliable protection against voltage surges, spikes and radio-frequency interference (RFI), which can damage circuitry and affect computer memory. The four-outlet, filtered power supply can be mounted easily, in any position, and a unique cable support organizes its power cords, draping them neatly to the rear of the work station.

new products

MICROCOMPUTER RESOURCE GUIDE FOR THE INDIVIDUAL INVESTOR

(reference book)
American Association of Individual Investors
612 N. Michigan Ave., Suite 317
Chicago, IL 60611
(312) 280-0170
$16.00

This resource book by Dr. Norman Nicholson is designed to fill the financial community's need for a computer user's guide to investing. As the editor of an investment newsletter, Dr. Nicholson brings his knowledge of financial-information services and time-sharing systems, as well as his understanding of microcomputing, to this 163-page publication.

BUSY BABY

(educational program)
Royale Software
Serpent JRd., DMHP #15
P.O. Box 351
Deerwood, MN 56444
(218) 534-3711
32K — diskette — $34.95

Busy Baby's main character is a wandering infant who must be guided through 26 levels of play — one for each letter of the alphabet. The game's primary purpose is to instill knowledge of the alphabet in the minds of young users, but it involves challenges that may also attract an older audience.

THE ART OF COMPUTER GAME DESIGN: REFLECTION OF A MASTER GAME DESIGNER

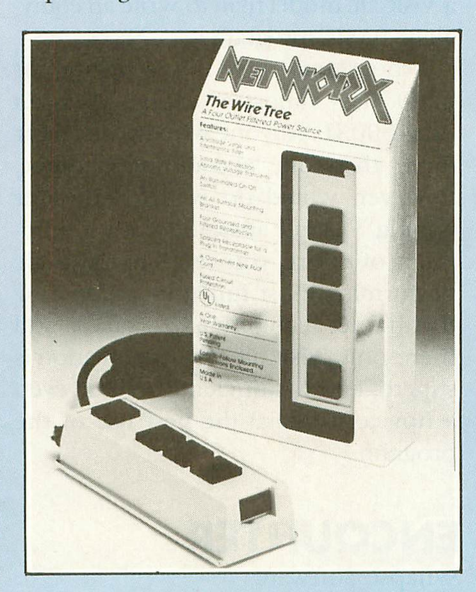
(book)
Osborne McGraw/Hill
2600 Tenth St.
Berkeley, CA 94710
(800) 227-2895
In CA — (800) 772-4077
$14.95

Although technical expertise comes in handy when designing games, the most important factor involved is creative vision. Chris Crawford, Atari programmer par excellence, teaches this and other noteworthy lessons in his guide to game design, which is based on his own programming discoveries and experiences.

Return the favor. When you call a manufacturer or supplier about a product you've seen advertised or otherwise mentioned in ANTIC, please tell them so. This will help us to continue to bring you the latest information about products that will make your Atari computer an even more valuable investment in the future. —ANTIC ED

KLEEN LINE CONDITIONER

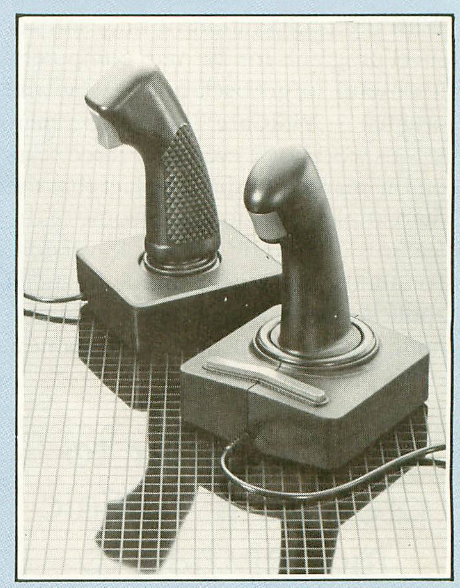
(computer-line conditioner)
Electronic Specialists, Inc.
171 So. Main St.
P.O. Box 389
Natick, MA 01760
(617) 655-1532
From $292.00–$977.00, depending on watt load



Designed to protect sensitive computer equipment, this series of portable computer-line conditioners is available for 250, 500, 1000 and 2000-watt loads. The conditioners deliver 120 volts at three percent regulation for 90- to 140-volt input variations. With three percent THD sine-wave output, the KLR series offers input-spike suppression, transformer-surge suppression, wide-band pre-filtering and isolated winding-line noise elimination. All models are available in decorator-styled cases.

TRIGA ELITE JOYSTICK and TRIGA COMMAND II

(joysticks)
Electra Concepts International Corp.
125 Wilbur Place
Bohemia, NY 11716
(516) 567-4783
Triga Elite — $23.95
Triga Command — $10.95



If your trigger finger is itching for a new toy, Electra has two new offerings for you.

The **Triga Elite**'s index-finger-operated firing button offers unmatched speed, while its independently operated continuous-fire bar lets you blast your way to higher scores.

Its sister, the **Triga Command II**, sports a contoured, "helicopter-style" handle with a diamond-cut texture, which makes the stick easy to grip and comfortable to use. Improved accuracy and speed are said to be possible due to the device's index-finger-controlled trigger action.

Both joysticks are compatible with the Atari 400/800 and 2600 systems, but only the Triga Elite can be used with the new XL computers.

New Products notices are solicited and compiled by Caitlin Morgan from information supplied by companies. **Antic** invites submissions, but assumes no responsibility for the accuracy of these notices nor the performance of these products.

product reviews

VISICALC

Atari, Inc.
P.O. Box 427
Sunnyvale, CA 94086
(408) 745-2000
$200.00, 48K — disk

Reviewed by Joseph Kattan

With the proliferation of new spreadsheet programs for the Atari computers, it's useful to go back and review the original — **VisiCalc**. It remains a marvelous workhorse even after three years on the market, despite the fact that the Atari implementation lacks some of the features offered by the Apple version. For almost any kind of home financial application, VisiCalc offers a programming environment that's unbeatable.

It's far easier to prepare a VisiCalc model than to write an equivalent BASIC program.

What exactly is a spreadsheet? It's a matrix made up of rows and columns. Each intersection of a row and a column is called a cell, and can contain text, a number, or a mathematical formula. Its ability to use formulas is what makes the electronic spreadsheet so powerful. For example, you can enter your budget expenditures in one column and use the cells in the adjacent column to calculate the percentage of your income devoted to each budget category. If you change any number in the budget column, the spreadsheet instantly displays new percentages for *all* categories based on the new number.

In addition to the standard arithmetic functions, VisiCalc finds the highest and lowest numbers in a list, calculates the average of numbers in a list, and looks up numbers in a table. You can use these features to perform sophisticated financial calculations at machine-language speed. Atari VisiCalc lacks the Boolean operators (AND, OR) and conditional statements (IF/THEN) found in other implementations, but for home use, it's more than enough.

VisiCalc isn't as easy to use as prepackaged home accounting programs, because you're required to design both the layout and the formulas used by the program. Because it is not pre-packaged, however, it's infinitely more powerful and flexible than such programs. You can use VisiCalc to balance your checkbook, keep track of credit card purchases, calculate your net worth, do your taxes — the possibilities are practically limitless. Using VisiCalc does require a minimum amount of programming skill, but it's far easier to prepare a VisiCalc model than to write an equivalent BASIC program.

Who should buy this program? At $200, it is almost as expensive as an Atari 800XL. Anyone who has need for more than one accounting package, however, would do well to consider buying VisiCalc instead. With a minimum of effort, you can have VisiCalc performing most functions offered by the home accounting packages, and then some. VisiCalc's documentation is superb, and is sufficient to guide even a novice to make the best use of the program.

ENCOUNTER

Synapse Software
5221 Central Ave., #200
Richmond, CA 94804
(415) 527-7751
$34.95, 32K — disk & cassette

Reviewed by James Trunzo

You've stumbled upon an ancient but technically sophisticated training enclave that has stood dormant for centuries. Once used to hone the battle skills of great space warriors, it has, upon your arrival, been reactivated. You must prove your mettle by completing the deadly training course — if you fail, you won't get another chance (unless you restart the game)!

Driving a tank-like vehicle, you maneuver about a pillar-strewn battlefield. Your adversaries in **Encounter** are of two types. Flying saucers move at random over the field, firing at you from any range. Drones, low-flying missiles that can home in on your position, are the second type of enemy you encounter. While you can, in theory, evade a saucer almost indefinitely (although this is not advisable!), you must destroy a drone immediately to avoid being destroyed yourself.

Once you've rid a level of enemies, a dimensional gate appears as a square black hole on the battlefield. Enter, and you'll find yourself rushing headlong into a breathtaking, multicolored meteor shower. If you pass through this stage without striking a meteor, you emerge into the next level. There are eight levels in all. Each new level introduces a different landscape, and adds to your enemies' repertoire of nasty tricks.

Encounter's use of graphics and sound is superlative.

Encounter's use of graphics and sound is superlative. Your point of view is that from your tank's front windshield, and the illusion of a three-dimensional landscape is made most convincing by the use of the pillars. The color combinations used in the different levels are pleasing to the eye.

To be perfectly honest, I was not impressed with Encounter at first blush. However, the game has grown on me since then, and I can recommend it quite highly. Encounter is one of the most challenging and exciting computer arcade games yet to have appeared in 1984.

THE ATARI USER'S ENCYCLOPEDIA

by Gary Phillips and Jerry White
The Book Company
11223 S. Hindry Ave.
Los Angeles, CA 90045
(213) 410-9466
$19.95

Reviewed by Fred Pinho

This Atari encyclopedia, the first of its kind, is a mixed bag. The first 44 pages are devoted to a tutorial for beginners. The encyclopedia section contains definitions of computer terms (both general and Atari-specific) and capsule descriptions of existing commercial software for Atari computers.

The beginner's tutorial contains a number of amply-documented program listings. By typing these in and running them, and by experimenting and observing, the beginner can learn a great deal about how computers work. The programs cover graphics, the keyboard, use of strings, and other topics. The tutorial combined with the reference material is good enough to constitute a stand-alone text on Atari BASIC. However, you may need to look elsewhere for advanced material on graphics.

This volume fills a real need for an overview of the world of Atari computing.

The technical entries in the encyclopedia section range in depth from a few cryptic words to a detailed description of a concept, complete with an illustrative program fragment. Thus, the level of computer expertise required to understand an entry varies widely. A reasonably complete memory map is included, along with a list of the changes that are implemented in the 1200XL. The software descriptions are quite brief, but to the point.

To give you an idea of the book's content, here's a list of a few of the entries: Accessories, ACTION!, Bus, CRC, DOS options, Floating-Point Representation, Hardware vs. Software, PRINT, Relative Addressing, and many, many more.

There are two useful appendices. The first lists software and hardware vendors of products for the Atari, and the second is a comprehensive list of Atari users' groups worldwide.

I'm somewhat ambivalent about this book. Having been a programmer for some time, I would have preferred a more sharply-focused work, with greater emphasis on technical and programming information. For the beginning-to-intermediate computerist, however, this volume fills a real need for an overview of the world of Atari computing. To those individuals, I recommend this book.

SOLO FLIGHT

MicroProse Software
10616 Beaver Dam Rd.
Hunt Valley, MD 21030
(301) 667-1151
$34.95, 48K — disk or cassette

Reviewed by Gordon Miles

The time has come for you to satisfy your yearnings for high-quality flight simulation. **Solo Flight** lets you pilot a 1930's-vintage monoplane for the sheer joy of flying, or make airmail deliveries in three states.

Your instrument panel fills the lower half of the screen. It comes complete with altimeter, speedometer, artificial horizon, throttle, flaps, pitch and climb rates, landing-gear controls, fuel gauge, air brakes, engine-temperature meter, compass and two navigational indicators. Wind and weather conditions are also shown. Use your joystick and several keys as controls.

The upper half of the screen reveals the view outside your monoplane's window. Solo Flight is not entirely a first-person simulation, however; from your cockpit you can view your monoplane from behind. When you bank or turn, the monoplane, not the horizon, changes position. This lessens the illusion of reality, but gives you a sense of immediate control over your plane's movements.

In Solo Flight, you soar over Kansas flatlands, travel through the skies above Washington's coastal mountains and cruise above Colorado's snowy eyries. Outlines mark airports, major mountains and towns, and extensive perspective- and color-gradation cues lend a sense of depth to the graphics. A shadow cast by the monoplane adds height cues. The illusion of looking out upon a large world is realized.

You'll enjoy the sensation of flying in this world. The scenery moves by rapidly, and the flight time is not overly long. If you like to tour, you can gaze out the side and rear windows. Although you can't roll or invert your plane, you can dive from 9000 feet and pull out at the last moment, buzz small towns or fly into clouds over mountainous terrain. You can also practice landings, create instrument malfunctions, fly through turbulence or depend solely on your instruments.

Solo Flight is the first high-quality flight simulation offered for Atari computers.

Flying in this world is also a good test of your navigational skills. If you don't know your left from your right, or if you can't read maps, Solo Flight can help you. It will also teach you the relationships between aeronautical factors such as pitch, flap lift and throttle power.

Soon, however, you'll tire of flying

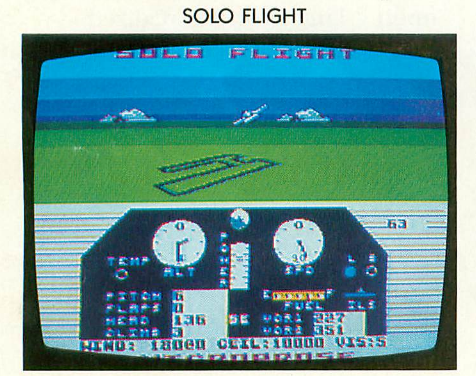
continued on next page

product reviews

aimlessly about, and will move on to Solo Flight's airmail delivery game. To play, you must deliver mail to five different airports during deteriorating weather conditions. Meanwhile, you're also hampered by occasional instrument malfunctions. This game is very challenging, particularly when your navigational aids go out, your engine overheats and clouds obstruct your view of the landmarks below. Fortunately, after every landing (or crash) your course is plotted on a map, so you can see where you went astray.

My review copy did not include night-flight or multiple-player options, but it was worth its price even without them. Solo Flight is the first high-quality flight simulation offered for Atari computers.

SOLO FLIGHT



THE SPY STRIKES BACK

Penguin Software
P.O. Box 311
Geneva, IL 60134
(312) 232-1984
$19.95, 32K — disk

Reviewed by Mark Cotone

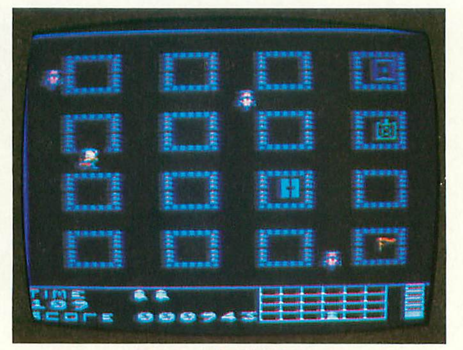
First came **The Spy's Demise**, a deceptively simple-looking game in which your task was to make several uninterrupted trips to the top of a 12-story building. So far, only four people have solved the puzzle at the heart of the game.

Now the sequel, **The Spy Strikes Back**, has been introduced. It's another game of espionage hide-and-seek, with another impossible code to crack.

The action takes place in the converted castle of Dr. Xavier Tortion, an international terrorist who intends to nuke a major population center. Your mission is to move through the castle, avoid guards, pick up and decipher pieces of coded information, and use this knowledge to find the bomb's hiding place. Pretty simple, eh?

Each of the castle's five floors is divided into 24 guarded sections. Each section fills one screen, and contains 16 rooms. You move your spy from screen to screen, trying to locate Allied spies who will provide you with necessary instructions.

You can easily learn the skills involved in moving your spy and eluding



THE SPY STRIKES BACK

the guards. But the decoding sequence will severely tax the amateur cryptographer. So, while arcade demons with well-honed reflexes may be able to traverse the castle's many sections unscathed, they may not have the patience required to unravel the secret code. On the other hand, puzzle-minded players may become bored, or frustrated, by the repetitive chore of travelling through the castle to pick up clues. The game's virtues tend to appeal to rather specialized audiences.

As a sales incentive, Penguin Software has announced The Spy Strikes Back Contest. "The first person in each state, Canadian province, or country to solve the coded puzzle will win $100 worth of software." I wonder if anyone will win.

MR. ROBOT AND HIS ROBOT FACTORY

Datamost
20660 Nordhoff St.
Chatsworth, CA 91311-2750
(213) 709-1202
$34.95, 32K — disk

Reviewed by Steve McLeod

Pinball Construction Set, move over! Here's another game that lets you build new computer game screens without ever touching the keyboard.

Mr. Robot is the hero of this action arcade game. His job is to roam the different levels of each screen (there are 22 on the disk), and collect the power pills buried in the floors. Each screen is

Robot Factory has more action in it than three or four of the leading maze games combined.

harder than the one before.

Your joystick is also a hero, because you can use it to design new levels — as many as you like. Among the types of building blocks available are ladders, escalators, fire poles, transporters, trampolines and treadmills.

You can also build floors out of time bombs. As Mr. Robot passes over them, the fuses light and they explode in less than a second. If Mr. Robot is positioned over one when it explodes, he's not only blown to Kingdom Come, but he also loses a life.

In short, **Mr. Robot and His Robot Factory** has more action in it than three or four of the leading maze games combined.

When you enter the Robot Factory, you're presented with a blank screen;

your building blocks are arrayed at the bottom of the screen. You can pick up an element and place it anywhere on the screen by using your joystick. And, if you like, you can easily "paint" an element over a large area. You also can play-test your screens, save them to disk, and string them together to create your own, personalized Mr. Robot saga.

The Robot Factory is easy to use, but the games aren't easy to win, particularly if you use time bombs and the alien fire monsters that follow Mr. Robot and try to gobble him up. He has a limited amount of time to complete each screen. Well-designed screens combine action and strategy to create a really frantic pace.

I played Mr. Robot for hours without tiring of the game. If a screen became a bit too familiar, I played another or designed my own. You may find that the best game screens are the ones you create.

MICROFILER

Microbits Peripheral Products
225 West 3rd St.
Albany, OR 97321
(503) 967-9075
$49.95, 16K — cartridge

Reviewed by George J. Adamson

As the ranks of computer owners grow, more and more people are discovering the wonders and intricacies of electronic filing, or data base management. Unfortunately, new users tend to be intimidated by the complexity of data base programs, and those who lack a disk drive have been cut off from the phenomenon completely.

Microbits Peripheral Products has remedied this situation somewhat with its new cartridge-based data base program, **MicroFiler**. The program works well with cassette-based systems, and eliminates the need for disk swapping on a one-drive system.

What makes MicroFiler so useful is its simplicity and versatility. You can use it to create mailing lists, inventories, club rosters, checkbook balances, and index-card files, and to print labels and lists.

The manual makes it easy for you to create a customized data base. Just type in labels wherever you want them on the screen, followed by data fields whose length you choose. By the way, a field can be defined as numeric, for use in computation.

The number of records that you can fit in one data base depends on the size of the fields in each record. In a 48K machine, I found that the use of a one-line, 36-character field in the record allows 1037 records, a two-line field allows 518 records, and a three-line field allows 352 records. When I created a three-line check-balancing data base that showed the date, a 28-character field for the payee, and the amount, I had room for 776 records.

> MicroFiler works well with cassette-based systems, and eliminates the need for disk swapping.

The process of entering data is simplicity itself. The "Retrieve" command displays a set of prompts — Forward, Backward, Rest, and Search. (You can search any field by entering a character string to be located.) Retrieve also lets you use the commands Change, Print, Delete, Sum (to add or subtract numeric fields), and Average (to find the average of each numeric field).

A unique feature of this program is that a file's size is limited to the computer's memory capacity. This allows the rapid retrieval and storage of data, but limits you to one file per disk. You can easily transfer your data-entry screen to a new disk, though. And cassette owners can store several files on a single cassette.

MicroFiler's limited file space makes it unsuitable for full-scale business use, but it is well-suited to the typical Atari hobbyist's needs. Because of its versatility, most users will find it unnecessary to buy additional data base programs.

THE COMPUTER TUTOR: ATARI

by Gary W. Orwig and William S. Hodges

Little, Brown Microcomputer Bookshelf
200 West St.
Waltham, MA 02154
(617) 890-0250
(800) 343-9204
$15.95

Reviewed by David Plotkin

The Computer Tutor: Atari is a collection of programs written both in Atari BASIC and Atari Microsoft BASIC, designed to provide computer-assisted instruction (CAI). While, in general, the programs don't make use of the Atari's special graphics and sound features, they are easy to enter and debug, work well, and offer some insight into BASIC programming techniques.

The Computer Tutor offers a wide range of instructional programs, including math, spelling, nations' capitals, memory tests, a story writer, and a series of simulations including the stock market, acceleration, and ballistics. All programs are presented both in Atari BASIC and Microsoft BASIC. Instructions for each program are printed in large, easy-to-read type, and listings are REMarked well to aid in understanding the program. Also, as a debugging aid, sample runs are included for each program. Suggestions for modifying the programs are also included.

The Computer Tutor presents quite a well-rounded sampling of CAI-type programs. As mentioned, the programs don't use graphics or sound, so you

continued on next page

product reviews

might wish to dress them up somewhat. Fortunately, the authors address this contingency well by including an appendix of graphics and sound routines. It's easy to supplement any of the BASIC programs with one or two of these subroutines. They're still not that fancy, but they are quite short. Anyway, who needs flashy graphics in CAI, especially at the cost of having to enter long, difficult listings. The Computer Tutor presents a well-balanced mix of interesting programs, and is a good buy for households with children and early teenagers.

TELLY TURTLE

Carousel Software, Inc.
877 Beacon St.
Boston, MA 02215
(617) 437-9419
$34.95, 24K — disk
$34.95, 16K — cassette

Reviewed by Vincent Puglia

Many of today's educational programs for preschoolers have two major drawbacks. Either they neglect the nonreader, or they fail to allow the child to explore on his or her own. Such is not the case with **Telly Turtle**, a programming language for drawing. By providing icons (pictographic symbols) and an open environment, the program encourages even the youngest computerist to learn programming, and much more, with minimal direction from an adult.

Telly Turtle is based on Logo's turtle graphics, but all manipulation is done with a joystick. The child selects a drawing function and a color from the row of icons at the bottom of the screen. To draw rectangles, squares and other figures, the child simply alternates between a straight-arrow icon and one of two turn arrows. Among the 11 icons on the first level, four indicate direction, four select colors, one "lifts" the pen from the drawing surface, and one is a Clear and Home symbol.

Telly Turtle's best feature is that it isn't easily outgrown by a child. As the user becomes familiar with the program, he or she graduates from simple commands that are executed immediately to fairly complex procedures that must be fully programmed before execution. There are four levels of complexity, each building on the preceding level. Level Two painlessly introduces the concept of looping. Level Three introduces programming in the indirect mode. The fourth level is almost a full-blown programming language, complete with nested loops, editing features, and I/O (input/output) commands.

As with all programming languages, Telly Turtle has limitations. However, you would be hard-pressed to find a language more suitable than this for teaching good programming techniques to your preschooler. If you buy this program, I strongly recommend that you let your child explore it on his or her own as much as possible. After all, every programmer has the right to learn that a program line that says "10 GOTO 10" won't go far.

Telly Turtle's best feature is that it isn't easily outgrown by a child.

PIE MAN

Penguin Software
P.O. Box 311
Geneva, IL 60134
(312) 232-1984
$19.95, 32K — disk
$19.95, 16K — cassette

Reviewed by Richard Herring

I'm sure you've seen the IBM commercial on TV in which a Chaplinesque character runs along a conveyor belt and tries to ice and box cakes before they reach the end of the line. Despite his frantic efforts, all of the cakes crash to the floor. The frustration Charlie feels approximates the feeling you get when you play **PIE MAN**.

Each time a whistle on the oven blows, a pie emerges onto a conveyor. Your baker must top each pie with whipped cream and a cherry (found in bins across the room), and must then carry it to the pie bin. During this time, of course, the conveyor keeps on rolling, and new pies appear.

While your baker attempts to save the pies, obstacles such as grease spills, flour sacks and a troublesome baker, who periodically runs through the kitchen with a precariously balanced load of pastries, appear. They block your baker's path between the bins and the conveyor.

PIE MAN is one of those rare games that does not duplicate another arcade or computer game.

Throughout the game, which lasts until seven pies have fallen off the conveyor, a series of seven melodies (from "Strawberry Blonde" to "Hot Time in the Old Town Tonight") entertain you. If you toggle the music off, the game's other sound effects remain in effect.

PIE MAN is one of those rare games that does not duplicate another arcade or computer game. Novelty and an amusing musical score are its strong points. Lasting play value, however, is not PIE MAN's forte. Everyone at my house enjoyed PIE MAN, but after several rounds we all decided to move on to a new game or back to an old favorite.

ABBY'S House of Discount Software

ATARI

SOFTWARE

ATARI

Atari Basic (CT) $ 29
Hangman (C) $ 7
States & Capitals (C) $ 7
European States & Capitals (C) $ 7
Kingdom (C) $ 7
Stock Analysis (D) $ 7
Bond Analysis (D) $ 7
Stock Charting (D) $ 7
MS Pac-Man (CT) $ 35
Pengo (CT) $ 35
Pole Position (CT) $ 35
Jungle Hunt (CT) $ 35
Joust (CT) $ 35
Eastern Front (CT) $ 33
Donkey Kong JR (CT) $ 35
Dig Dug (CT) $ 33
Atari Microsoft II (D&CT) $ 69
Atari Logo (CT) $ 79
Pilot (CT) $ 69
Assembler Ed (CT) $ 49
Atari Macro Assembler (D) $ 69
Atari Writer (CT) $ 75
Visicalc (D) $169

AVALON-HILL 4/800 1200 XL

North Atlantic Convoy (D) $ 16
VC (D) $ 16
Plane Miners (D) $ 16
Conflict 2500 (C) (D) $ 16
Voyager (C) (D) $ 16
Galaxy (C) (D) $ 16
Space Station Zulu (C) (D) $ 16
Knockout (C) $ 12
RD Racer/Bowler (C) $ 10
Shootout at OK Galaxy (C) (D) $ 13
Guns at FT. Defiance (C) (D) $ 16
Tank Arcade (C) $ 10
Moon Patrol (C) $ 16
Vorrak (D) $ 17
Gypsy (C) (D) $ 12
Flying Ace (C) (D) $ 17
Lords of Karma (C) (D) $ 16
Empire of the Empire (C) (D) $ 21
Acquire (C) (D) $ 12
Stock & Bonds (C) (D) $ 13
Draw Poker (C) (D) $ 10
Controller (C) (D) $ 21

BRODERBUND

AE (D) $ 17
Apple Panic (C) (D) $ 17
Bank Street Writer (D) $ 45
Choplifter (CT) (D) $ 19
David's Midnight Magic (D) $ 19
Drol (D) $ 19
Genetic Drift (C) (D) $ 17
Ladyrinth (C) (D) $ 19
Lode Runner (D) $ 19
Matchboxes (C) (D) $ 15
Operation Whirlwind (D) $ 29
Serpentine (C) (D) (CT) $ 17
Sea Fox (D) $ 17
Sky Blaser (D) $ 15
Stellar Shuttle (C) (D) $ 15
Track Attack (D) $ 17

CBS

K-razy Shoot Out (CT) $ 9
K-razy Kritters (CT) $ 9
K-Star Patrol (CT) $ 9
K-dos (D) $ 15

COMPUTER MAGIC

Kayos (CT) $ 9

DATA SOFT

Canyon Climber (C) (D) $ 15
Clowns & Balloons (C) (D) $ 15
Graphic Master (D) $ 25
Graphic Generator (D) $ 15
Moon Shuttle (C) (D) $ 13
O'Riley's Mine (C) (D) $ 13
Pacific Coast Hwy (C) (D) $ 15
Sands of Eqypt (D) $ 19
Shooting Arcade (C) (D) $ 13
Zaxxon (D) $ 19

ELECTRONIC ARTS

Pinball Construction Set $ 39
Music Construction Set $ 39
Archon $ 39
D-Bug $ 34

EPYX

Jumpman (D) $ 29
Jumpman JR. (CT) $ 29
Pitstop (CT) $ 29
Gateway to Temple of Aphasi (CT) $ 29
Dragon Riders of Pern (D) $ 29
Starfire/Fire One (D) $ 29
Temple of Aphasi (D) (C) $ 29
Upper Reaches of Aphasi (D) (C) $ 17
Curse of RA (D) (C) $ 17

GAME STAR

Star Bowl Football (C) (D) $ 17

GAMMA

Soccer (D) (C) $ 15
Hockey (D) (C) $ 15

GEBELLI

Pathfinder (CT) $ 13
Doc Goodlode's Cavern (CT) $ 13
Embargo (CT) $ 13

IMAGIC

Demon Attack (C) $ 11

INFO COM

Enchanter (D) $ 39
Witness (D) $ 39
Planetfall (D) $ 39
Infidel (D) $ 39
Sorcerer (D) $ 39
Zork I, II, III (D) $ 29
Sea Stalker (D) $ 39
Starcross (D) $ 29
Deadline (D) $ 35

IN HOME

Gork (D) (C) $ 8
Sentinel (D) (C) $ 8

MUSE

Castle Wolfenstein (D) $ 17

ROKLAN

Gorf (D) $ 17
Wizards of Wor (D) $ 17
Deluxe Invaders (D) $ 17

SIERRA ON-LINE

Crossfire (D) (CT) (C) $ 17
Frogger (D) $ 19
Golf (C) $ 17
Jaw Breaker (C) (CT) (D) $ 17
Lunar Leeper (D) $ 17
Mouskatter (D) $ 17
Marauder (D) $ 17
Threshold (D) $ 17
Wall War (D) $ 17

SPECTRUM COMPUTER

Galactia Chase (C) (D) $ 9

SPINNAKER

Alphabet 200 (CT) $ 27
Delta Drawing (CT) $ 29
Facemaker (CT) $ 27
Fraction Fever (CT) $ 27
Kids on Keys (CT) $ 27
Kinder Comp (D) (CT) $ 22
Story Machine (D) $ 20
Riddles & Rhymes (D) $ 20
Aerobics (D) $ 35
Trans (D) $ 29
Snoopertroops I (D) $ 29
Snoopertroops II (D) $ 29

SWIFTY

Haunted Hill (C) (D) $ 11
Space Shuttle (D) $ 11
Trivia-Trek (D) $ 11
Disk Inventory (D) $ 11

SYNAPSE

Chicken (C) (D) $ 17
Claim Jumper (C) (D) (CT) $ 17
Dodge Racer (D) $ 17
Nautilus (C) (D) $ 17
Picnic Paranoia (C) (CT) (D) $ 12
Protector II (C) (D) $ 17
Shamus (C) (D) $ 17
Slime (C) (D) $ 12
Fort Apocalysis (C) (D) $ 27
Pharoh's Curse (C) (D) $ 27
Blue Max (C) (D) $ 29

THORN EMI

Kickback (C) $ 7
Darts (C) $ 7
Snooker (C) $ 7
Dominoes (C) $ 7
Hickory Dickory (C) $ 7
British Heritage 1, 2 (C) $ 7
Super Cubes (C) $ 7

HARDWARE

ATARI

1200 XL $189
600 XL $169
800 XL $259
1027 $299
1025 $325
410 $ 49
1010 $ 79
1050 $349
Bit 3 80 Column $239
Trak A & Z Disk Drive $369
Indus GT Disk Drive $399

MONITORS

Sakata 100 C Monitor $259
NEC 1260 $ 89
NEC 1201 $139
NEC 1215 $249
Gorilla Green $ 89
Gorilla Amber $ 99
Amdek 300 $139
Amdek 300A $159
Amdek Color IT $319

DISKS

Nashua SSSD, Box of 10 $ 17
Elephant SSSD, Box of 10 $ 19
Elephant SSDD, Box of 10 $ 24
Elephant DDDD, Box of 10 $ 31

DUST COVERS

600 XL $ 8
800 XL $ 8
1200 XL $ 8
1025 $ 8
1010 $ 6

Koala Touch Pad $ 75

(C) TAPE (D) DISC (CT) CARTRIDGE

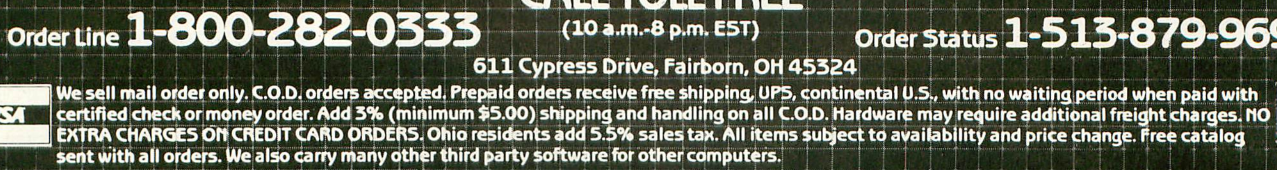
FREE DISKETTE WITH EACH PURCHASE!

ATARI

CALL TOLL FREE

Order Line 1-800-282-0333 (10 a.m.-8 p.m. EST) Order Status 1-513-879-9699

611 Cypress Drive, Fairborn, OH 45324





SHORTCUTS TO SUCCESS

A guide to Atari macro graphics

by THOMAS McNAMEE

SYNOPSIS

This article explains the use of macros in assembly-language programming, and supplies a library of macros for graphics programming. To use the accompanying programs, you must use MAC/65, a macro assembler from Optimized Systems Software. The macros and demonstration program will run on any Atari computer.

Quick! Identify the following language:

```
0250 GRAPHICS 3
0260 COLOR 1
0270 SETCOLOR 4,0,4
0280 SETCOLOR 0,4,14
0290 PLOT 5,5
0300 DRAWTO 5,15
0310 DRAWTO 15,5
0320 DRAWTO 15,5
0330 DRAWTO 5,5
```

It's not BASIC, of course — that would have been too easy. But, when executed, it draws the same pink square on a grey background that BASIC would have drawn. This is, or will become, assembly language. More precisely, it is a series of commands, called macro calls, which can be expanded and compiled into executable 6502 machine code by a powerful tool called a macro assembler.

This article will explain macro assembly and demonstrate the ways in which macros can be used to simplify program generation. It will also provide you with a graphics library that can be used as easily as the above listing suggests. The MAC/65 macro assembler from Optimized Systems Software (OSS) must be used to run the macros and the demonstration program included in this article.

WHAT IS MACRO ASSEMBLY?

There is an unwritten rule in commercial software development: Never write the same code twice. Because macros are reusable, tested modules, they allow you to stick to this guideline. Also, because numeric and string parameters can be passed on to them, their execution can be affected during compilation without making it necessary to change the code.

I should explain at this point that there is an important distinction between run-time and compile-time behavior. Run-time behavior is, quite simply, what the code does during the process of execution. Once the assembler produces machine code, run-time behavior is frozen. For example, an LDA #$FF always loads 255 into the accumulator. Compile-time behavior, on the other hand, concerns the disposition of operations, operands, and pseudo-ops as they go through the assembly process.

Take a calculation, for example: LDA #$C0 + 1 becomes LDA #$C1 in machine code. Compile-time behavior depends to a significant degree on the assembler being used.

Consider this simple macro:

```
.MACRO LOADX
.IF %1<255
  LDX #%1
  .ELSE
  LDX %1
  .ENDIF
.ENDM
```

All keywords that start with " . " are commands to the assembler. They only affect compile-time behavior, and do not appear in the machine code after assembly. For example, a two- or three-byte LDX instruction would be compiled from the above listing.

Several commands are introduced in LOADX. The command %1 means "parameter 1." This parameter is passed in the macro call as shown in the following example:

```
0250 LOADX 7
```

The seven replaces every occurrence of %1 in the macro. First, the .IF statement determines if it's less than 255. The .IF/.ELSE/.ENDIF assembler commands all make conditional assembly possible. In this case, the <255 test is true, so LDX #7 is compiled. If it were false, the assembler would assume that the passed parameter was a non-Zero-Page memory location, and the LDX would be compiled to a three-byte, absolute-addressed instruction.

As a macro call is assembled, it goes through a process called expansion. Listing 1 shows an example of this. Note

continued on next page

that each time you call a macro, it is included in the program in its expanded form. As a result, if you use large macros, your program begins to consume huge amounts of memory. This is an important consideration when you're using macros. Because of this, it's usually more practical to incorporate a subroutine call to a single macro into your programs.

MACRO ADVANTAGES

The use of macros is advantageous primarily because passed parameters can be used to customize a macro for any application. The GRAPHICS macro in Listing 3 will accept any argument that the BASIC GRAPHICS command will; it will also use a memory location if the argument is nine or greater. Another advantage is that once you've tested a macro, you have a known, useful piece of software at your disposal. Thus, a macro library, such as the one in Listing 3, constitutes a collection of reusable, tested routines that can be used any time they're needed in a program.

LIBRARY AND EQUATE FILES

Listing 3 contains macros that simulate many BASIC graphics commands. Once you've typed it in and saved it, you can use .INCLUDE to incorporate it into any of your programs. The use of macro libraries adds a great deal of efficiency to the process of program development, because it saves a good deal of composition and testing time.

It's also helpful to use function-oriented EQUate files in your programs These files give standard Atari mnemonic labels to commonly-used locations, and they can also be .INCLUDEd during assembly. Listing 2 contains equates used by Listing 3, and should be .INCLUDEd before assembling the graphics macros.

THE GRAPHICS.LIB FILE

The commands included in Listing 3 simulate Atari BASIC's commands as closely as possible. When compiled, however, they execute many times faster than BASIC. The following list describes each of these macros in brief:

GRAPHICS — Allows you to call Graphics Modes 0–8. However, you must allocate enough memory to permit the S: driver to open the screen. To do this, first calculate the memory requirements for the screen, display list and text window. Subtract this quantity from the address of the top of RAM, and store the result in APPMHI, which is equated in Listing 2. This macro uses all registers; upon exit, the Y-register contains the status.

COLOR — Stores the selected color-register number in a spare byte that is equated in Listing 2. Only the accumulator is used.

POSITION — Positions the cursor at the selected screen location. Only the accumulator is used. Note that the passed parameters must be literals, not memory locations.

PLOT — Plots a point in the selected COLOR at the X and Y coordinates, which are passed as %1 and %2.

DRAWTO — Draws a straight line from the last point plotted to the passed coordinates. All registers are used.

SETCOLOR — Sets the selected register to the indicated hue and color. All registers are used.

DEMONSTRATION PROGRAM

Listing 4 is a short program that demonstrates the use of the above graphics macros, as well as the use of .INCLUDEd files. Before you assemble it, make sure that you have all four .INCLUDE files on a disk in your drive. GREQU.M65 is Listing 2 and GRAPHICS.LIB is Listing 3.

These listings and the application program detail the use of each graphics macro. With some experience, you can create your own library, thereby saving yourself a fair amount of programming and testing time. A good starting point would be to code two routines: SAVEREG and RESTORE. These should allow you to preserve all registers while making a subroutine jump or macro call.

Thomas McNamee is a software engineer for ManTech International in Alexandria, Virginia. He programs in FORTH, C, BASIC and 6502 assembly language, and has written for a number of computer publications.

Listing 1

```
               0350       COLOR   1
     =0001     M         .IF %1<16
3000 A901      M           LDA #%1
               M           .ELSE
               M           LDA %1
               M           .ENDIF
3002 8DF502    M         STA SPARE1
               M         .ENDM
               0360       SETCOLOR   4,
0,4
     =0001     M         .IF %1<16
3005 A204      M           LDX #%1
               M           .ELSE
               M           LDX %1
               M           .ENDIF
     =0000     M         .IF %2>16
               M           LDA %2
               M           ASL A
               M           ASL A
               M           ASL A
               M           ASL A
               M           .ELSE
3007 A900      M           LDA #%2*16
```

```
                    M            .ENDIF
     =0001          M          .IF %3<16
3009 A004           M            LDY #%3
                    M            .ELSE
                    M            LDY %3
                    M            .ENDIF
300B 9DC402         M          STA COLOR0,X
300E 98             M          TYA
300F 290E           M          AND #$0E
3011 18             M          CLC
3012 7DC402         M          ADC COLOR0,X
3015 9DC402         M          STA COLOR0,X
                    M          .ENDM
```

Listing 2

```
0100 ;FILENAME: GREQU.M65
0110     .TITLE "GRAPHICS EQUATE FILE"
0120     .PAGE "GRAPHICS EQUATES"
0130 ;
0140 ; CURSOR
0150 ROWCRS = $54      ; FULL SCREEN
0160 COLCRS = $55
0170 TXTROW = $0290    ; SPLIT SCREEN
0180 TXTCOL = $0291
0190 ;
0200 ; SCREEN MEMORY
0210 SAVMSC = $58      ; FULL SCREEN
0220 TXTMSC = $0294    ; SPLIT SCREEN
0230 ;
0240 ; GRAPHICS MODE
0250 DINDEX = $57      ; FULL SCREEN
0260 TINDEX = $0293    ; SPLIT SCREEN
0270 ;
0280 ; SCREEN MARGINS
0290 LMARGN = $52
0300 RMARGN = $53
0310 ;
0320 ; DISPLAY LIST
0330 VDSLST = $0200    ; VECTOR FOR DLI
0340 SDLSTL = $0230    ; START OF DL
0350 WSYNC = $D40A
0360 ;
0370 ; MISC.
0380 BOTSCR = $02BF    ; # LINES IN SCREEN (4 OR 24)
0390 SPARE1 = $02F5    ; SPARE BYTES
0400 SPARE2 = $02F6
0410 SPARE3 = $02F7
0420 SPARE4 = $02F8
0430 SPARE5 = $02F9
0440 ATACHR = $02FB    ; USED BY FILL & DRAW
0450 ;
0460 ; CHARACTER SET
0470 CHBAS = $02F4     ; SHADOW
0480 CHBASE = $D409    ; HARDWARE
0490 CHACT = $02F4     ; CHAR. MODE REG
0500 ;
0510 ; COLORS
0520 ; SHADOW REGISTERS
0530 COLOR0 = $02C4
0540 COLOR1 = $02C5
0550 COLOR2 = $02C6
0560 COLOR3 = $02C7
0570 COLOR4 = $02C8
0580 ; HARDWARE REGISTERS
0590 COLPF0 = $D016
0600 COLPF1 = $D017
0610 COLPF2 = $D018
0620 COLPF3 = $D019
0630 COLBK = $D01A
0640 ;
0650 ; INTERRUPT CONTROL
0660 NMIEN = $D40E     ; NMI ENABLE
0670 NMIST = $D40F     ; NMI STATUS
0680 NMIRES = $D40F    ; NMI RESET
0690 ; BIT: 7   6   5   43210
0700 ;      DLI VBI RESET .....
0710 ;
0720 ; SCROLLING
0730 HSCROL = $D404
0740 VSCROL = $D405
0750 ;
0760 ; CIO COMMANDS
0770 CDRAW = $11
0780 CFILL = $12
0790 ;
```

Listing 3

```
0100 ;FILENAME: GRAPHICS.LIB
0110     .TITLE "GRAPHICS LIBRARY"
0120     .PAGE "GRAPHICS MACRO"
0130 ;
0140     .IF .NOT .DEF ROWCRS
0150       .ERROR "GREQU.M65 missing!"
0160       .ENDIF
0170     .IF .NOT .DEF CPALOC
0180       .ERROR "SYSEQU.M65 missing!"
0190       .ENDIF
0200 ;
0210 ;MACRO: GRAPHICS
0220 ;
0230 ;FORM: GRAPHICS N
0240 ;If N is <9, then N is a memory
0250 ;location, else N is a literal.
0260 ;
0270     .MACRO GRAPHICS
0280 ;
0290     JMP @CONT
```

continued on next page

```
0300 @AX1
0310      .BYTE 0
0320 @AX2
0330      .BYTE 0
0340 @CONT
0350       XIO  12,6,0,0,"S:"
0360      .IF %1<9
0370        LDA #%1
0380        .ELSE
0390        LDA %1
0400        .ENDIF
0410      STA @AX2
0420      AND #$F0
0430      EOR #$10
0440      ORA #$0C
0450      STA @AX1
0460       OPEN  6,@AX1,@AX2,"S:"
0470      .ENDM
0480      .PAGE "COLOR, POSITION & PLOT MACROS"
0490 ;
0500 ;MACRO: COLOR
0510 ;
0520 ;FORM: COLOR N
0530 ;If N<16, then N is literal.
0540 ;Else N is a memory location.
0550 ;
0560      .MACRO COLOR
0570      .IF %1<16
0580        LDA #%1
0590        .ELSE
0600        LDA %1
0610        .ENDIF
0620      STA SPARE1
0630      .ENDM
0640 ;
0650 ;MACRO: POSITION
0660 ;
0670 ;FORM: POSITION X,Y
0680 ;X and Y must be literals, not
0690 ;memory locations
0700 ;
0710      .MACRO POSITION
0720      LDA # <%1
0730      STA COLCRS
0740      LDA # >%1
0750      STA COLCRS+1
0760      LDA #%2
0770      STA ROWCRS
0780      .ENDM
0790 ;
0800 ;MACRO: PLOT
0810 ;
0820 ;FORM: PLOT X,Y
0830 ;X and Y must be literals, not
0840 ;memory locations
0850 ;
0860      .MACRO PLOT
0870       POSITION  %1,%2
0880       @CH  6
0890      LDA #CPBINR
0900      STA ICCOM,X
0910      LDA #0
0920      STA ICBLEN,X
0930      STA ICBLEN+1,X
0940      LDA SPARE1
0950      JSR CIO
0960      .ENDM
0970      .PAGE "DRAWTO & SETCOLOR MACROS"
0980 ;
0990 ;MACRO: DRAWTO
1000 ;
1010 ;FORM: DRAWTO X,Y
1020 ;X and Y must be literals as in
1030 ;POSITION
1040 ;
1050      .MACRO DRAWTO
1060       POSITION  %1,%2
1070      LDA SPARE1
1080      STA ATACHR
1090       @CH  6
1100      LDA #CDRAW
1110      STA ICCOM,X
1120      LDA #CCLOSE
1130      STA ICAUX1,X
1140      LDA #0
1150      STA ICAUX2,X
1160      JSR CIO
1170      .ENDM
1180 ;
1190 ;MACRO: SETCOLOR
1200 ;
1210 ;FORM: SETCOLOR REG,HUE,LUM
1220 ;Any parameter <16 is considered
1230 ;a memory location
1240 ;
1250      .MACRO SETCOLOR
1260      .IF %1<16
1270        LDX #%1
1280        .ELSE
1290        LDX %1
1300        .ENDIF
1310      .IF %2>16
1320        LDA %2
1330        ASL A
1340        ASL A
1350        ASL A
1360        ASL A
1370        .ELSE
1380        LDA #%2*16
1390        .ENDIF
```

```
1400        .IF %3<16
1410          LDY #%3
1420          .ELSE
1430          LDY %3
1440          .ENDIF
1450        STA COLOR0,X
1460        TYA
1470        AND #$0E
1480        CLC
1490        ADC COLOR0,X
1500        STA COLOR0,X
1510        .ENDM
1520 ;
1530 ;******* END OF GRAPHICS.LIB ***
```

Listing 4

```
0100 ;FILENAME: GRTEST.M65
0110        .OPT NO LIST ;Don't list ]INC
LUDE files
0120 ;
0130 ;These files are on MAC/65 disk.
0140        .INCLUDE #D:SYSEQU.M65
0150        .INCLUDE #D:IOMAC.LIB
0160 ;
0170 ;See text for these files.
0180        .INCLUDE #D:GREQU.M65
0190        .INCLUDE #D:GRAPHICS.LIB
0200 ;
0210 GRMEM = 460       ;Bytes of memory
for GR. 3
0220 ;
0230        *=  $3000
0240 ;
0250        .OPT LIST
0260 ;
0270 ;Set top of applications memory
0280        LDA # <$BFFF-GRMEM
0290        STA APPMHI
0300        LDA # >$BFFF-GRMEM
0310        STA APPMHI+1
0320 ;
0330 ;This looks like BASIC...
0340         GRAPHICS  3
0350         COLOR  1
0360         SETCOLOR  4,0,4
0370         SETCOLOR  0,4,14
0380         PLOT  5,5
0390         DRAWTO  5,15
0400         DRAWTO  15,15
0410         DRAWTO  15,5
0420         DRAWTO  5,5
0430 ;
0440 ;Loop until SYSTEM RESET
0450 END JMP END
0460        .OPT NO LIST
```

SCROLL YOUR WAY TO THE TOP continued from page 45

Listing 1

```
1000 REM SCROLLING EXAMPLE
1100 REM BY CHRIS CHABRIS
1200 REM ANTIC MAGAZINE
1600 POKE 559,0:DIM SCRP$(29),SCRN$(29
),DF$(59)
1700 REM See Listing 3 for following m
achine language subroutines:
1800 RESTORE 1900:FOR L=1 TO 29:READ B
:SCRP$(L)=CHR$(B):NEXT L
1900 DATA 104,104,104,133,203,162
2000 DATA 0,189,4,6,24,101
2100 DATA 203,157,4,6,144,3
2200 DATA 254,5,6,232,232,232
2300 DATA 224,36,208,235,96
2400 RESTORE 2500:FOR L=1 TO 29:READ B
:SCRN$(L)=CHR$(B):NEXT L
2500 DATA 104,104,104,133,203,162
2600 DATA 0,189,4,6,56,229
2700 DATA 203,157,4,6,176,3
2800 DATA 222,5,6,232,232,232
2900 DATA 224,36,208,235,96
3000 RESTORE 3100:FOR L=1 TO 59:READ B
:DF$(L)=CHR$(B):NEXT L
3100 DATA 104,201,4,208,39,104
3200 DATA 104,10,10,10,10,170
3300 DATA 104,104,157,66,3,104
3400 DATA 157,69,3,104,157,68
3500 DATA 3,104,157,73,3,104
3600 DATA 157,72,3,32,86,228
3700 DATA 152,133,212,169,0,133
3800 DATA 213,96,170,104,104,202
3900 DATA 208,251,169,22,133,212
4000 DATA 169,0,133,213,96
4100 REM Reserve memory; load map and
character set files:
4200 M=PEEK(106)-12:MAP=M*256:CP=M+8:C
H=CP*256:POKE 106,M-1:POKE 756,CP
4300 OPEN #1,4,0,"D1:SCROLL.CHR":Q=USR
(ADR(DF$),1,7,CH,512):CLOSE #1
4400 OPEN #1,4,0,"D1:SCROLL.MAP":Q=USR
(ADR(DF$),1,7,MAP,960):CLOSE #1
4500 POKE 708,112:POKE 709,38:POKE 710
,230:POKE 711,66:POKE 712,8
4600 REM Build new display list on Pag
e 6:
4700 DL=1536:POKE DL+0,112:POKE DL+1,1
12:POKE DL+2,112
4800 FOR L=3 TO 36 STEP 3:POKE DL+L,71
:MMEM=MAP+40*(L/3-1)
4900 MH=INT(MMEM/256):ML=MMEM-MH*256:P
OKE DL+L+1,ML:POKE DL+L+2,MH:NEXT L
5000 POKE DL+39,65:POKE DL+40,0:POKE D
L+41,6:POKE 560,0:POKE 561,6
5100 POKE 559,34:X=9:Y=5
5200 REM Main program loop follows:
5300 H=0:V=0:J=STICK(0):IF J=15 THEN 5
300
5400 IF J=6 OR J=7 OR J=5 THEN H=1
5500 IF J=10 OR J=11 OR J=9 THEN H=-1
```

continued on next page

```
5600 IF J=9 OR J=13 OR J=5 THEN V=1
5700 IF J=10 OR J=14 OR J=6 THEN V=-1
5800 XN=X+H:YN=Y+V
5900 IF ((XN<9 OR XN>29) AND H<>0) OR
((YN<5 OR YN>17) AND V<>0) THEN 5300
6000 OFS=40*V+H:IF OFS>0 THEN Q=USR(AD
R(SCRP$),OFS):GOTO 6200
6100 Q=USR(ADR(SCRN$),-OFS)
6200 X=XN:Y=YN:GOTO 5300
```

TYPO TABLE

Variable checksum = 360661

Line num range	Code	Length
1000 - 2400	AV	412
2500 - 3600	VD	325
3700 - 4500	IG	543
4600 - 5300	XP	503
5400 - 6200	FC	388

Listing 2

```
1000 REM FILE GENERATOR
1200 REM BY CHRIS CHABRIS
1300 REM ANTIC MAGAZINE
1700 DIM H$(2),DAT$(96),HEX$(23):HEX$=
"@ABCDEFGHI#######JKLMNO"
1800 REM Process character set data:
1900 OPEN #1,8,0,"D1:SCROLL.CHR":RESTO
RE 2900:CS=0:FOR LINE=2900 TO 3900 STE
P 100:GOSUB 2400:NEXT LINE:CLOSE #1
2000 REM Process map data:
2100 OPEN #1,8,0,"D1:SCROLL.MAP":RESTO
RE 4100:CS=0:FOR LINE=4100 TO 6000 STE
P 100:GOSUB 2400:NEXT LINE:CLOSE #1
2200 ? "FILES GENERATED":END
2300 REM Subroutine to convert data an
d write to disk:
2400 READ DAT$,C:FOR L2=1 TO LEN(DAT$)
 STEP 2:H$=DAT$(L2,L2+1)
2500 D=0:FOR I=1 TO 2:D=D*16+ASC(HEX$(
ASC(H$(I))-47))-64:NEXT I:CS=CS+D
2600 PUT #1,D:NEXT L2:IF C<>CS THEN ?
"BAD DATA AT LINE # ";LINE:END
2700 RETURN
2800 REM Hexadecimal data for characte
r set file:
2900 DATA 00000000000000001070400DC740C
183010180818302030181018 1CCC6C38000010
3038EB6E060000000010D878180C18,2155
3000 DATA 10180E1F190C0C180000E0B33A0E
0400000 0C3C76E38181010180 8F9633E0000FF
FFFFFFFFFFFFFFF0E4FDFFFFF6F3F8,7858
3100 DATA 01033FFEE4800000B0E0C0000000
0000003C664242424200557F2E3A3E77770000
637F2222227763005E524C7F4C1221,11221
3200 DATA 0000000000000000000000000000
00000000000000000000000000000000000000
000000000000000000000000000000,11221
3300 DATA 00040E2072071038207 0F820081C
3E08040E1F041038 7C100000103 87CFE100000
2070F800081C3E081C3E002070F800,13705
3400 DATA 000010387CFE000000A050081C3E
7F000094B600D81B406CFFF3C18000000000FF
DF8F1F0604000 0FFFFFFFEBE1C0800,18091
3500 DATA 0F1F0F0700000000FFFFFFFF9F0F
070F0F070F1F17070 30F3F7F3F1F0F1F070F07
070FDF7F4F070F0000000001030 90F,20351
3600 DATA 0F1F7F3F7FFFFFFF180C060E9FFF
FFFF000000 90F6FFFFFF0000004 0E2C7EFFF00
000000 60F0F8F0F0E0F0F9FFFFFFFF,27394
3700 DATA F0F8FCF8FCFEFCF8F0E0C0E0F0F8
F0E0E0F0F8F1FBFFFCF8F0E080C080000000FF
FFFFFFFEFCF6F00000000000000000,36103
3800 DATA 0000000000000000000000000000
00000000000000000000000000000000000000
00000000000000000A045025022810,36292
3900 DATA 00110A040044281000442810 0011
0A04101 87CFFFE3A781000000000000000000,3
7469
4000 REM Hexadecimal data for map file
:
4100 DATA CACACACACACACACACACACACACACA
CACACACACACACACACACACACACACACACACACACA
CACACACACACACACACA0A0A0A0A0A0A,8352
4200 DATA 0A0A0A0A0A0A0A0A0A0A0A0A0A0A
0A0A0A0A0A0A0A0A0A0A0A0A0A0A0A0A0A0ACACA
0A0A0A0A3421222322232125 0A0A0A,9429
4300 DATA 0A0A0A3421232225 0A0A0A342123
250A0A0A0A0A0ACACA0A0A3421335E5C00
0099BB24250A0A0A0A3199 00D026,11574
4400 DATA 0A0A34335CCE242221232225 0A0A
0ACACA0A0A305C5CCF5C5E000000BB242125 0A
0A0A2F2C2D2C2A0A0A2F2E5E5D5D5E,14315
4500 DATA 5C0000260A0A0ACACA0A34335C5E
5D5E5D5D0000009A9B2423250A0A0A0A0A0A0A
0A0A2F2D2C2E005C5E0024250A0ACA,16721
4600 DATA CA0A30BC5E005E025C5E5C5E0000
00000027 0A0A0A0A342321250A0A0A0A300000
5D5E5C24250ACACA0A30BB06070704,19090
4700 DATA 07050 05DCE5E5C0000260A0A0A0A
3000E0260A0A0A0A31BB0000005E5C270ACACA
0A0B0C0DBD00000002005E5D5D5E07,21508
4800 DATA 07280A0A342133002 92A0A0A0A34
33BC0000005D292A0ACACA0A30E00000000000
06070 75D5E5C5E00270A0A315D0000,23581
4900 DATA 270A0A0A30BC00000000005E260A
0ACACA0A2F2E00000009A9B005E5C000000000
260A0A315D0000260A34232233 0000,25470
5000 DATA 0000005D24250ACACA0A0A2F2E00
9B9A9B9A000000005C5C00242325320907 0728
0A30000000000000000000005C270ACA,27587
5100 DATA CA0A0A0A30999A9B3E9A9B000000
5C5E0000E024335C00BB24213300 9A9B000000
0000005E260ACACA0A0A0A2F2D2E99,30772
5200 DATA 029A9B990000005E5C0000000000
00000000000009B9A9A9B000005D292A0ACACA
0A0A0A0A3207 03CE9A9B0000005D,33348
5300 DATA 5E00292C2D2C2E5E5C009B9A292D
2C2E5E5C00E0270A0ACACA0A0A0A0A30999B
9A9A00000000005E5C5D270A0A3207,36325
5400 DATA 0500009B260A0A2F2C2E5E292A0A
0ACACA0A0A0A0A2F2E999B000000000000005E
5C24222321335C020000292A0A0A0A,38314
```

```
5500 DATA ØA2F2D2AØAØAØACACAØAØAØAØAØA
ØA2F2D2EØØØØØØ5FØØØØØØ5C5D5DCFØØ5D5CØ2
5DØØ27ØAØAØAØAØAØAØAØAØAØAØACA,40301
5600 DATA CAØA34222125ØAØAØA2F2EØØØØØØ
ØØØØØØ5E5D5D5E5C5D5CØ15DBB26ØAØA3425ØA
342225ØAØAØACACAØA3ØBCØØ242325,42792
5700 DATA ØAØA31ØØØØØØ292C2D2E5D5EØ8Ø7
Ø8Ø7Ø4Ø7Ø527ØAØA2F2AØA31DF28ØAØAØACACA
ØA31BBBCØØBB26ØAØA3ØDØ292B2AØA,45245
5800 DATA ØA2F2E5DØ2ØØØ2BCBC292B2AØAØA
ØAØAØA2F2B2AØAØAØACACAØA2F2D2C2B2C2AØA
ØA2F2D2AØAØAØAØAØA2F2C2B2C2B2D,47308
5900 DATA 2C2AØAØAØAØAØAØAØAØAØAØAØAØA
ØACACAØAØAØAØAØAØAØAØAØAØAØAØAØAØAØAØA
ØAØAØAØAØAØAØAØAØAØAØAØAØAØAØA,48238
6000 DATA ØAØAØAØAØACACACACACACACACA
CACACACACACACACACACACACACACACACACACACA
CACACACACACACACACACACACACACACA,56590
```

TYPO TABLE

Variable checksum = 151281

Line num range	Code	Length
1000 – 2300	NG	507
2400 – 3100	IX	574
3200 – 3600	WQ	540
3700 – 4200	QQ	542
4300 – 4700	SE	540
4800 – 5200	PK	540
5300 – 5700	SL	540
5800 – 6000	AT	324

Listing 3

```
; ASSEMBLY LANGUAGE LISTING
; BY CHRIS CHABRIS
; ANTIC MAGAZINE
        ORG   $6000
;These subroutines are all relocatable
;Routines to scroll display, callable
;from BASIC as follows:
;    Q=USR(ADR(SCRP$),OFS) or
;    Q=USR(ADR(SCRN$),-OFS)
;Where: OFS is the offset value
OFS     EQU   $00CB     ;Storage of offset
                        ;to scroll display
DL4     EQU   $0604     ;Fourth byte of
                        ;display list
                        ;(address lobyte)
SCRP    PLA             ;This scrolls with
                        ;a positive offset
                        ;value
        PLA
        PLA
        STA   OFS       ;Keep offset for
                        ;later use
        LDX   #00       ;In BASIC,
                        ;"FOR L=0..."
                        ;initialize index
LOOP1   LDA   DL4,X     ;Get lobyte of
                        ;LMS address
        CLC
        ADC   OFS       ;Add the offset
        STA   DL4,X     ;And replace with
                        ;the new value
        BCC   CONT1     ;If no carry,
                        ;continue on
        INC   DL4+1,X   ;Carry set, so
                        ;increment hibyte
CONT1   INX
        INX
        INX             ;Add three to the
                        ;index register
        CPX   #36       ;If X=36, we have
                        ;completed all
                        ;12 lines
        BNE   LOOP1     ;If not, go back
                        ;& do the next one
        RTS             ;If so, return to
                        ;BASIC
SCRN    PLA             ;This (almost
                        ;identical) is for
                        ;negative OFS
        PLA
        PLA
        STA   OFS
        LDX   #00
LOOP2   LDA   DL4,X
        SEC             ;These two lines
                        ;are different to
                        ;allow sub-
        SBC   OFS       ;traction of OFS
        STA   DL4,X
        BCS   CONT2     ;Same for this and
                        ;following line
        DEC   DL4+1,X
CONT2   INX
        INX
        INX
        CPX   #36
        BNE   LOOP2
        RTS
; Routine to load or save data
; callable from BASIC as follows:
; Q=USR(ADR(DF$),IOCB,CMD,ADDRESS,NUM)
; Where: IOCB is the Input/Output
```

continued on next page

```
; channel being used
;   CMD is 7 for READ or 11 for WRITE
;   ADDR is the address to take data
;        from or move data to
;   NUM is number of bytes to transfer
CIOV    EQU  $E456     ;Central Input/
                       ;Output vector
ICCOM   EQU  $0342     ;IOCB #0:
                       ; command byte
ICSTA   EQU  $0343     ; status byte
ICBAL   EQU  $0344     ; buffer address
                       ;     (lo-hi)
ICBLL   EQU  $0348     ; buffer length
                       ;     (lo-hi)
USRARG  EQU  $00D4     ;Address of USR
                       ;function return
                       ;argument
DF      PLA            ;Number of
                       ;parameters passed
                       ;by BASIC
        CMP  #04       ;Is it four as it
                       ;should be?
        BNE  ERROR     ;If No return with
                       ;variable Q=22
        PLA            ;Hibyte of IOCB
                       ;number - discard
        PLA            ;Lobyte of same
        ASL  A         ;Multiply it by 16
                       ;for ease of use
        ASL  A
        ASL  A
        ASL  A
        TAX            ;Save it for later
                       ;use in X-register
        PLA            ;Hibyte of
                       ;command - discard
        PLA            ;Lobyte of command
        STA  ICCOM,X   ;Put it in com-
                       ;mand byte of
                       ;IOCB #X/16
        PLA            ;This is hibyte
                       ;of the buffer
                       ;address
        STA  ICBAL+1,X   ;Into the IOCB
                         ;you go!
        PLA            ;Lobyte of buffer
                       ;address
        STA  ICBAL,X   ;goes in also
        PLA            ;Hibyte of buffer
                       ;length
        STA  ICBLL+1,X
        PLA              ;and lobyte of
                         ;same go
        STA  ICBLL,X     ;into proper
                         ;IOCB,
EXEC    JSR  CIOV      ;Now CIO will
                       ;perform the I/O
CHECK   TYA            ;Get completion
                       ;status from
                       ;Y-register
        STA  USRARG    ;And pass it back
                       ;to BASIC
        LDA  #00
        STA  USRARG+1    ;Zero-out hibyte
                         ;just in case
        RTS            ;Return to BASIC,
                       ;Q=Error number
ERROR   TAX            ;Save incorrect
                       ;# of arguments
                       ;passed
LOOP3   PLA            ;and pull them off
                       ;the stack one by
                       ;one
        PLA
        DEX
        BNE  LOOP3     ;Go do next one
                       ;if any left
        LDA  #22       ;ERROR- 22
                       ;indicates
                       ;incorrect # of
                       ;args.
        STA  USRARG    ;Let BASIC know
                       ;about it
        LDA  #00
        STA  USRARG+1    ;Make sure the
                         ;error number
                         ;is <256!
        RTS            ;BASIC regains
                       ;control,
                       ;unsatisfied
        END
```

ADVERTISERS

This is provided as a convenience and as a courtesy to advertisers. ANTIC does not guarantee accuracy or comprehensiveness.

listing conventions

Table Information

Our custom font listings represent each ATASCII character as it appears on the video screen. You generate some characters by a single keystroke, for example, the regular alphabet. Others require a combination or sequence of keystrokes. In this table, ESC means *press and release* the escape key before pressing another key. CTRL or SHIFT means *press and hold* the control or shift key while simultaneously pressing the following key.

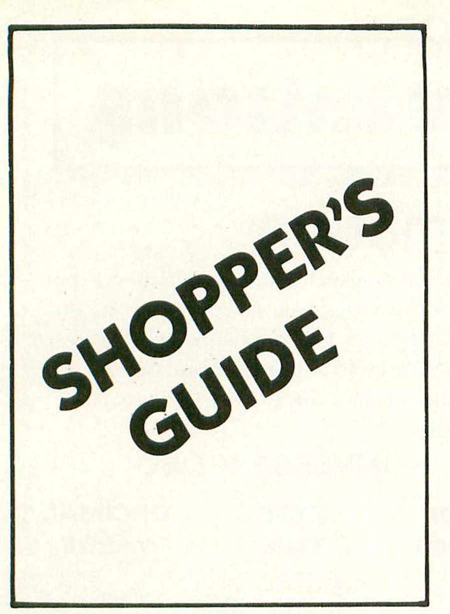
The Atari logo key (人) "toggles" inverse video for all alphanumeric and punctuation characters. Press the logo key once to turn it on; press again to turn it off. In the XL line there is no logo key; inverse video is controlled by a key on the function row. Decimal values are given as reference, and correspond to the CHR$ values often used in BASIC listings.

NORMAL VIDEO

FOR THIS	TYPE THIS	DECIMAL VALUE
	CTRL ,	0
	CTRL A	1
	CTRL B	2
	CTRL C	3
	CTRL D	4
	CTRL E	5
	CTRL F	6
	CTRL G	7
	CTRL H	8
	CTRL I	9
	CTRL J	10
	CTRL K	11
	CTRL L	12
	CTRL M	13
	CTRL N	14
	CTRL O	15
	CTRL P	16
	CTRL Q	17
	CTRL R	18
	CTRL S	19
	CTRL T	20
	CTRL U	21
	CTRL V	22
	CTRL W	23
	CTRL X	24
	CTRL Y	25
	CTRL Z	26
	ESC ESC	27
	ESC CTRL −	28
	ESC CTRL =	29
	ESC CTRL +	30
	ESC CTRL *	31
	CTRL .	96
	CTRL ;	123
|	SHIFT =	124
	ESC SHIFT CLEAR	125
	ESC DELETE	126
	ESC TAB	127

INVERSE VIDEO

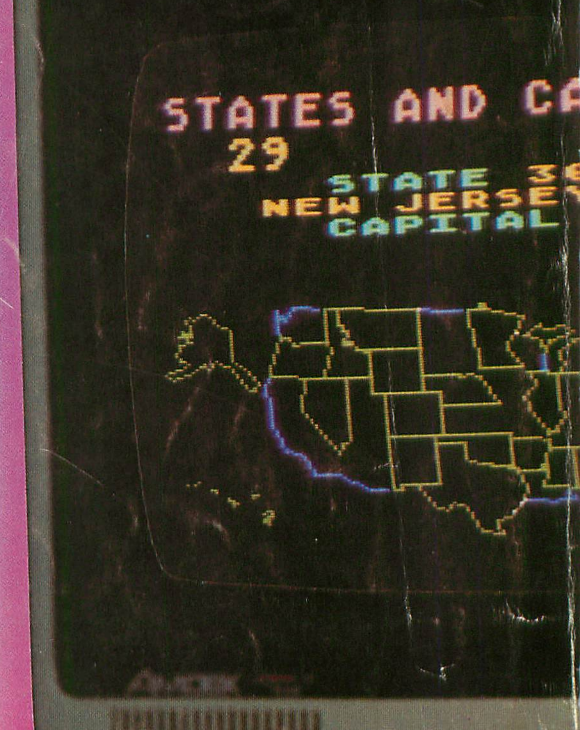
FOR THIS	TYPE THIS	DECIMAL VALUE
	人CTRL ,	128
	人CTRL A	129
	人CTRL B	130
	人CTRL C	131
	人CTRL D	132
	人CTRL E	133
	人CTRL F	134
	人CTRL G	135
	人CTRL H	136
	人CTRL I	137
	人CTRL J	138
	人CTRL K	139
	人CTRL L	140
	人CTRL M	141
	人CTRL N	142
	人CTRL O	143
	人CTRL P	144
	人CTRL Q	145
	人CTRL R	146
	人CTRL S	147
	人CTRL T	148
	人CTRL U	149
	人CTRL V	150
	人CTRL W	151
	人CTRL X	152
	人CTRL Y	153
	人CTRL Z	154
	ESC SHIFT DELETE	156
	ESC SHIFT INSERT	157
	ESC CTRL TAB	158
	ESC SHIFT TAB	159
	人CTRL .	224
	人CTRL ;	251
	人SHIFT =	252
	ESC CTRL 2	253
	ESC CTRL DELETE	254
	ESC CTRL INSERT	255


SHOPPER'S
GUIDE


Vol. 1 No. 6 Tools
Vol. 2 No. 1 Games
Vol. 2 No. 2 Communication
Vol. 2 No. 3 Data Base Survey
Vol. 2 No. 4 Adventure Games
Vol. 2 No. 5 Graphics
Vol. 2 No. 6 Education
Vol. 2 No. 7 Sports Games
Vol. 2 No. 8 Sound & Music
Vol. 2 No. 9 Buyer's Guide
Vol. 2 No. 10 Printers
Vol. 2 No. 11 Personal Finance
Vol. 2 No. 12 International
THEY'RE GOING FAST!!
Limited numbers of ANTIC's back issues are still available! Get them while they last! Mail your order today!
$4.00 each USA
$5.00 each Foreign
(Must remit U.S. funds by check or draft on U.S. banks.)
☐ Vol. 1 No. 6
☐ Vol. 2 No. 1
☐ Vol. 2 No. 2
☐ Vol. 2 No. 3
☐ Vol. 2 No. 4
☐ Vol. 2 No. 5
☐ Vol. 2 No. 6
☐ Vol. 2 No. 7
☐ Vol. 2 No. 8
☐ Vol. 2 No. 9
☐ Vol. 2 No. 10
☐ Vol. 2 No. 11
☐ Vol. 2 No. 12
_____ Back issues @ $ _____ each. Total $ _____
Name _____
Address _____
please print
City _____
State _____ Zip _____
☐ Visa ☐ MasterCard
Account # _____
Name _____ Expiration Date _____
please print
Signature _____
Allow 8 weeks for delivery
ANTIC™ Publishing Inc. 524 Second Street San Francisco, CA 94107

BUILT FOR YOUR ATARI'S FUTURE...
AMDEK TOUGH!

Put the futuristic Amdek disk drive and 3″ diskette to work with your Atari® now. Get tomorrow's performance today. It's small...quiet...convenient to use...and "user tough." Competitively priced for use in the home, business or classroom.



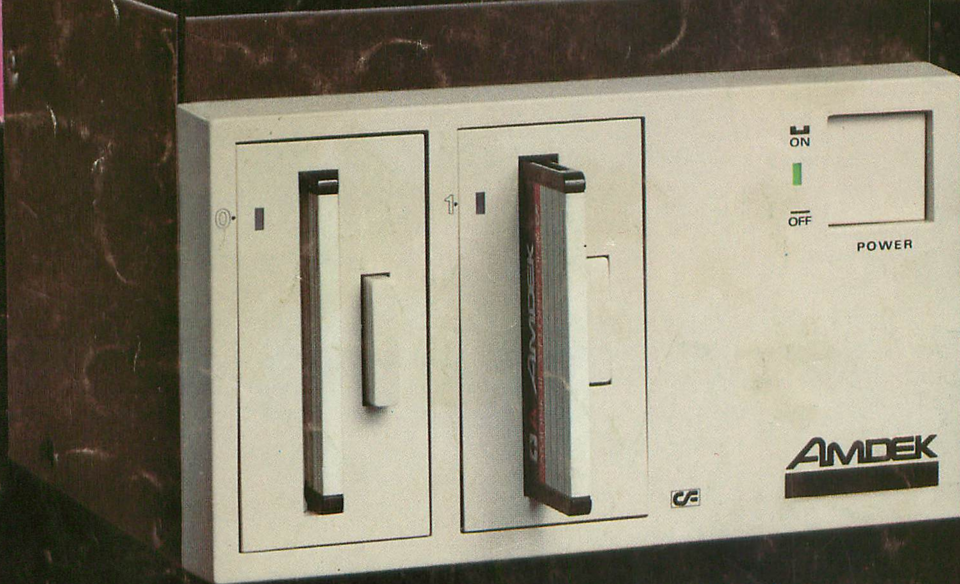
With the Amdek AMDC I or AMDC II disk drive (single or dual drive), you get superior performance. Hook up your printer or plotter direct. Use in conjunction with any 5¼″ disk drive to boot other Atari-compatible software.

Amdek diskette cartridges are virtually destruction-proof. They will support both single and double density recording. You get 180,000 characters of storage capacity on each side. You get expanded use...more games...more programs...more capabilities. Many software packages are already available on these 3″ diskettes. And, more are planned.

As you've come to expect from Amdek, you get a complete package—operating software, cable, instruction manuals, and product support.

Call Your Local Computer Dealer Today...and make your Atari the most powerful ever.

Atari® is a registered trademark of Atari, Inc.





2201 Lively Blvd. • Elk Grove Village, IL 60007 • (312) 364-1180 • TLX 25-4786
REGIONAL OFFICES: Southern Calif. (714) 662-3949 • Northern Calif. (408) 866-8134 • Texas (817) 498-2334 • Denver (303) 794-1497